AD700033

MEMORANDUM
RM-5940-PR
SEPTEMBER 1969

A MODEL FOR TARGETING STRIKES IN AN LOC NETWORK

R. D. Wollmer and M. J. Ondrasek

PREPARED FOR:
UNITED STATES AIR FORCE PROJECT RAND

The RAND Corporation
SANTA MONICA • CALIFORNIA

49

MEMORANDUM
RM-5940-PR
SEPTEMBER 1969

A MODEL FOR TARGETING STRIKES IN AN LOC NETWORK

R. D. Wollmer and M. J. Ondrasek

This research is supported by the United States Air Force under Project RAND—Contract No. F44620-67-C-0045—monitored by the Directorate of Operational Requirements and Development Plans, Deputy Chief of Staff, Research and Development, Hq USAF. Views or conclusions contained in this study should not be interpreted as representing the official opinion or policy of the United States Air Force.

The RAND Corporation

Published by The RAND Corporation

PREFACE

This Memorandum describes a model for targeting strikes against a lines-of-communication (LOC) network and evaluating the effect of carrying out the strategy that results. It does not handle vehicle interdiction, which is treated in [4]. The model was developed while the authors were members of a special RAND-AFGOA (Operations Analysis Office, Hq USAF) joint Southeast Asia study group. Both RAND and AFGOA are currently applying the model and will report their results to the Air Force as part of a follow-on effort of the group's activities. The model generalizes an earlier one developed by Durbin [1], based on the mathematics of Wollmer [6] and applied in RAND and AFGOA studies. The mathematical basis for the model of this Memorandum may be found in [7].

While the model was originally intended for immediate use by the RAND-AFGOA study group and later internal use by both RAND and AFGOA, several other agencies have expressed interest in it as an aid in their own research. Among these are the Assistant Chief of Staff, Studies and Analysis, Hq USAF; Army-Air Force Intratheater Transportation Requirements Study Group; Directorate of Force Planning Analysis, Office of the Chief of Staff, U.S. Army; and the Boeing Company. It should also prove useful to other groups concerned with targeting LOCs, evaluating LOC interdiction effectiveness, predicting the value of LOC interdiction tactics, or with constructing and repairing transportation systems.

M. J. Ondrasek is a Captain in the Directorate of Operations, Hq USAF (AFXOXSB).

SUMMARY

This Memorandum presents a model for developing and evaluating a targeting strategy against an opposing force's lines of communications (LOCs). There are several ways to measure how effective a strategy is in reducing the usefulness of the enemy's LOCs. Among these are reduction in throughput (flow), increase in cost of meeting throughput requirements, and combinations of the two.

LOCs are represented by a network of directed arcs (line segments) and nodes (points). Arcs may represent road, rail, or waterway segments, or points of transshipment. They are characterized by their beginning and ending nodes, upper and lower flow bounds, interdicted and uninterdicted unit flow costs, repair times, repair costs, and probabilities that attempted strikes result in successful interdiction. Nodes are junctions where two or more arcs intersect.

The model operates in daily cycles, the user specifying the number of days and strikes per day. At the start of each day, certain interdicted arcs are returned to operation (as specified by their repair times and last time struck); the capacities and unit flow costs of these arcs are restored to uninterdicted values. Then the appropriate number of strikes are targeted, one at a time, against the LOC arcs by the second algorithm in Ref. [7]. The algorithm chooses targets based on the cost and time involved in repairing the arc and on how the arc's disablement will reduce network effectiveness. The procedure is optimal when one strike is targeted, and approximates optimality when more than one strike is targeted. At the end of each strike, total throughput and the necessary cost to the LOC user of achieving this throughput are produced as output.

Two other outputs are available to the model user, and either or both may be requested after each strike or at the end of the day. The first is a detailed status of the entire network, including individual arc flows; the second is a profile of total flow versus required user cost.

ACKNOWLEDGMENTS

The authors benefitted significantly from several helpful discussions with E. P. Durbin, and from the computer programming support provided by Durbin and Jo Ann Lockett.

CONTENTS

I. INTRODUCTION

This Memorandum presents a model for targeting strikes against an opponent's lines of communication (LOCs) and evaluating their effectiveness in reducing his ability to deliver supplies. Effectiveness of strikes may be measured by reduction in throughput, cost of operation to the LOC user, or combinations of these. Operating costs may be dollar costs, manpower or personnel costs, or other appropriate measures.

Single-mode LOCs or multimode LOCs with transshipment between modes can be represented by a network of arcs and nodes. Arcs may represent road, rail, or waterway segments, or points of transshipment between modes. Nodes are junctions where two or more arcs intersect. Arcs are characterized by their beginning and ending nodes, lower and upper bounds on flow, and unit flow costs. In this model, an arc's upper bound on flow and unit flow cost are parameters that are functions of the number of strikes targeted against the arc. These functions depend for their arguments on the values of these parameters when the arc is struck successfully, when unstruck or struck unsuccessfully, and on the probability it has been struck successfully; it is the latter quantity that varies with the number of strikes.

Many network flow problems can be formulated to find a minimum cost circulation flow by artificial devices.* Among these are maximizing flow from a source node to a sink node, meeting a required flow between two nodes at minimum operating cost, and combinations of these two. It is assumed that the effectiveness of the LOCs to the user may be formulated in this context. The effectiveness of a strike is measured in terms of how it reduces LOC usefulness to its user.

The programmed model operates in daily cycles. Each day a given number of strikes are targeted against the LOC arcs after certain arcs from among those previously struck are returned to their uninterdicted

*As will be shown later, this cost, obtained partly by artificial devices, often differs from actual operating costs.

condition. Arcs are selected as targets on the basis of what strikes on them contribute towards reducing LOC effectiveness to its user. Specifically, the arc chosen for the next strike is the one that will maximize the repair cost plus the product of the repair time and the cost increase of a minimum cost circulation flow.

II. NETWORK REPRESENTATION OF LOCs

SINGLE MODE

A single-mode system of transportation routes such as roads or waterways can be represented by a network of nodes and directed arcs. Specifically, for a highway system, a node is the intersection of two or more roads, and an arc represents a road segment joining two nodes. For a one-way segment, the beginning node is the point where the traffic enters the segment and the ending node the point where traffic leaves it. A two-way road segment can be represented by two arcs, one in each direction. Similar representations hold for railroads, waterways, and other modes. As an example, consider the network below.

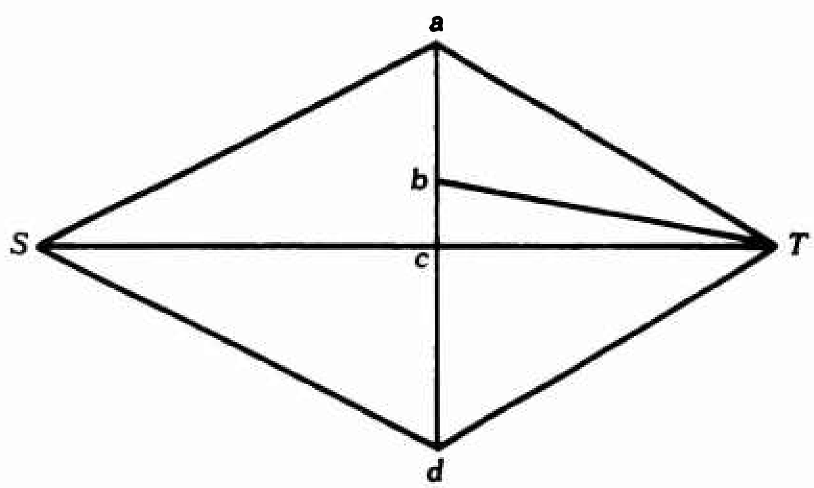

Fig. 1 -- A simple network

Figure 1 might represent a road system with segments joining locations S and a, a and T, b and T, c and T, d and T, S and c, S and d, a and b, b and c, and c and d. All roads are two-way, and hence each segment is represented by two arcs.

Each arc has certain parameters: flow capacity (the upper bound on flow), lower bound on flow, and cost per unit flow. As an example, let us suppose that the parameters for the arc in Fig. 1 joining S and

a and in the direction of S to a, designated by the symbol (S,a), are as in Table 1.

Table 1

PARAMETERS FOR ARC (S,a) IN FIG. 1

Parameter	Value
Flow	6
Upper Bound	8
Lower Bound	0
Cost	3

In Table 1, the actual flow from S to a is 6. This can be tons/day, trucks/hour or any other appropriate measure. Other values of flow are possible, but only between the inclusive limits of zero and 8. Lower bounds normally are zero, as in this example; but as we shall see, there is sometimes a reason for having them greater than zero. The upper bound here could represent the road's physical capacity to handle traffic. The cost per unit flow of 3 is in whatever units are used for the example; possibilities include dollars per ton moved from S to a, man-hours per vehicle traveling from S to a, and vehicle hours. The total cost for the flow of 6 on this arc would be 3 times 6, or 18. Note that one would normally expect that except for flow, all parameters for the arc in the opposite direction (a,S) would be identical to those of (S,a).

MULTIMODE WITH TRANSSHIPMENT

A system with several modes of transportation can be represented by first constructing a separate network for each mode and then connecting these networks at transshipment points. For example, let us take the simple railroad network of Fig. 2, together with the road network of Fig. 1. The points S,T,c, and d in Fig. 2 represent the same points as these same symbols in Fig. 1. Suppose transshipment is allowed only at nodes S and d. Then the combined road-rail network may be represented as in Fig. 3. Note that c and c' (the two nodes representing point c) and T and T' are not connected. This is because flow entering point c

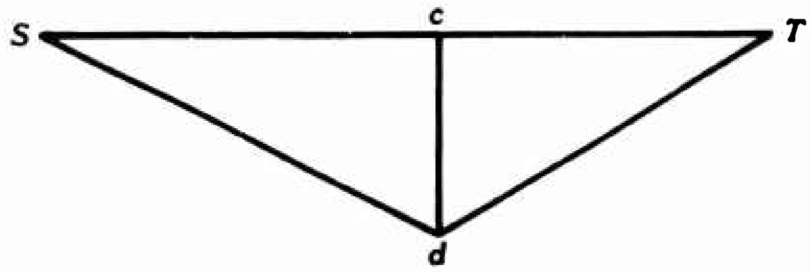

Fig. 2 -- Network representation of railroad routes

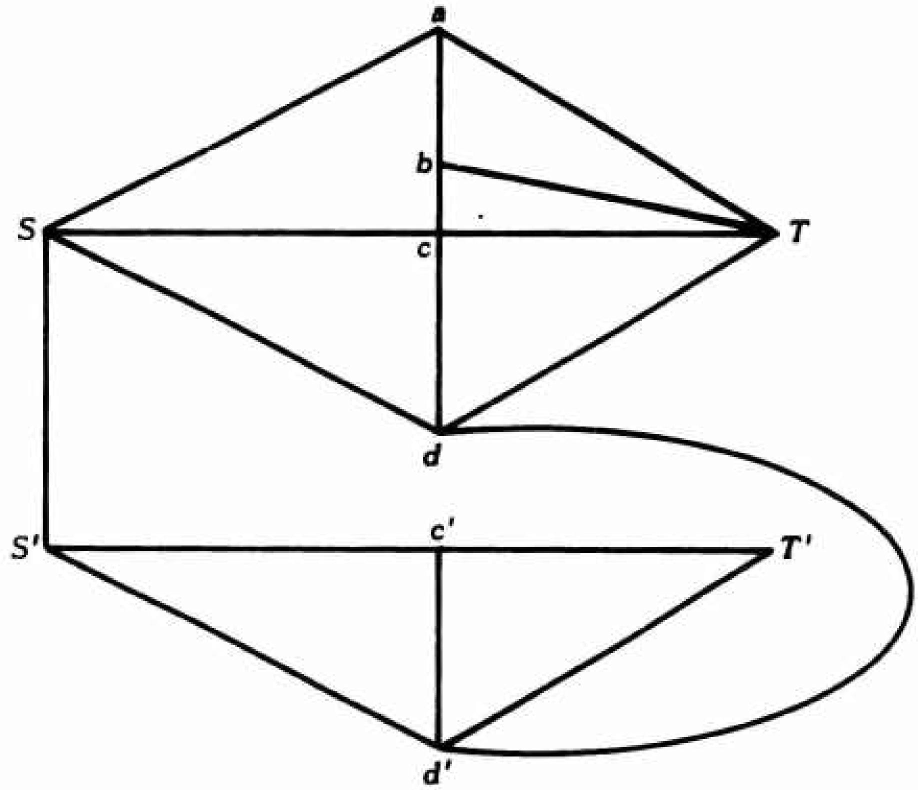

Fig. 3 -- Combined road-rail network of Fig. 1 and Fig. 2

or point T by one mode cannot leave by another. This also means that c and c' (and similarly T and T') cannot be represented by one node, if the network is to have the property that flow entering a node via a particular arc can leave that node via any arc emanating from that node. The flow, upper bound, lower bound, and cost on arc (d,d') represent the amount transshipped from road to rail at that point, capacity of the transshipment facility, lower bound on use of facility (usually

zero), and cost per unit of flow transshipped. Flow on (d',d) represents the amount transshipped from rail to road, and other parameters have the same meaning as in arc (d,d').

SOURCES, SINKS, AND FLOWS

Very often one is interested in sending units of flow from one node to another. As an example, one may wish to transport equipment from a supply depot to a battlefront. In such cases the node the flow originates from is called the source and the node it terminates at is called the sink.

As an example, suppose in Fig. 1 the user wished to send units of flow from S to T. Then S would be the source and T would be the sink. If only one unit were sent from S to T, a possible route is the path S,a,b,T; this would be represented by a unit of flow on arcs (S,a), (a,b), and (b,T), and zero flow on all other arcs. A flow of one unit over an alternate route, S,c,T would be represented by one unit on (S,c), one on (c,T), and zero units on all other arcs. Note that in each case, the net flow out of S and the net flow into T both equal one, while the net flow into or out of any other node is zero. In general networks, if x units were to be sent from a source node to a sink node, then any flow pattern accomplishing this would have net flows of x units out of the source, x units into the sink, and zero into (or out of) any other node. The reverse also follows; specifically, any flow pattern having net flows of x units out of the source, x units into the sink, and zero units of net flow into or out of any other node, can be broken up into units of flow on routes from source to sink totaling x. In addition to these routes, however, there may be cycles or closed loop routes.

In actual flow situations, there may be several sources and several sinks. Such problems may be handled in the LOC model provided any source may supply any sink. Of course, this condition is always met if there are several sources and one sink or one source and several sinks. The reason for this condition is that it allows one to convert, through artificial devices, the multisource, multisink problem to a single-source, single-sink problem, and technically we require but one source and one

sink. The way this conversion is accomplished is to connect, with artificial arcs, all sources to an artificial super source and all sinks to an artificial super sink. For example, suppose in the network of Fig. 1 that a and S were sources and d and T were sinks. Then this network could be represented by the single-source, single-sink network of Fig. 4.

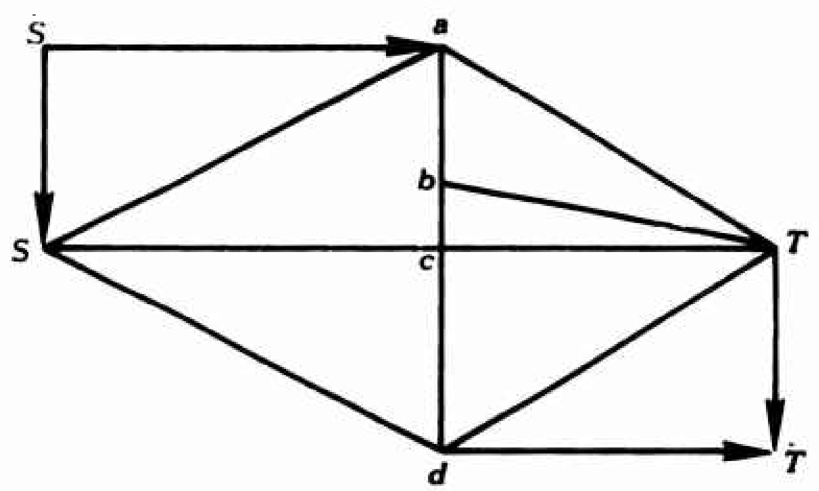

Fig. 4 -- Equivalent network to double source and double sink network of Fig. 1. The arrows indicate the direction of the arc. Lines without arrows indicate arcs in both directions.

The artificial or super source, $\bar{S}$ (in Fig. 4), would be the source and $\bar{T}$ the sink. Flow on the artificial arc $(\bar{S},a)$ equals the rate that goods leave the true source, a. The lower bound on $(\bar{S},a)$ would normally be zero; the upper bound might be essentially infinite or, if it represented a facility such as a warehouse, could be the maximum supply rate at that facility; and cost, if not zero, could be equal to the unit cost of using facilities represented by a. The same interpretation may be given to the parameters of $(\bar{S},S)$. An artificial arc connecting a sink to the artificial super sink node carries a flow equal to the quantity per unit time entering the sink it represents, a lower bound equal

to zero or the sink demand, an upper bound essentially infinite or equal to sink demand, and a cost either zero or equal to the unit cost of using the sink facility. Note that such a formulation as in Fig. 4 assumes any source can supply any sink in any proportion. Thus this formulation would not be valid if it were required that flow emanating from source a must terminate at sink d.

III. USER EFFECTIVENESS AND CIRCULATION FLOWS

TRANSPORT USER EFFECTIVENESS OF LOCs

As will be shown in the next section, strike effectiveness is based quite heavily on the degree to which it reduces the usefulness of the LOCs to the user. User effectiveness of LOCs may be measured in several ways. Among these are the following:

1. Maximum flow from one node (source) to another (sink).
2. Minimum cost of meeting a required source-sink flow.
3. Minimum cost of required flow if required flow can be met; otherwise, the least cost of maximizing flow.

The first of these is realistic when the user is physically limited by the ability of his transport system to handle traffic. The second is reasonable when he is not limited by his transport system but is constrained by a resource such as dollars or man-hours, or merely desires to use resources efficiently. The third is one of several possible combinations of the first two, and applies when the user does not know whether or not he is constrained by his transport system and also desires to use resources efficiently.

CIRCULATION FLOWS

A minimum-cost circulation flow pattern in a network can be found with the "Out-of-Kilter" algorithm [3]. Such a network problem requires minimizing total cost subject to the constraint that all arc flows are between their upper and lower bounds, and the flow entering a particular node must equal the flow leaving it. Not so obvious is the fact that each user effectiveness criterion discussed above can be formulated as a minimum-cost circulation flow problem.

Converting source-sink flows to circulation flows is accomplished as follows. Connect a universal arc directed from the sink to the source and assign it a flow equal to that of the total source-sink flow. Letting S be the source and T the sink in Fig. 1, this network would be redrawn as in Fig. 5.

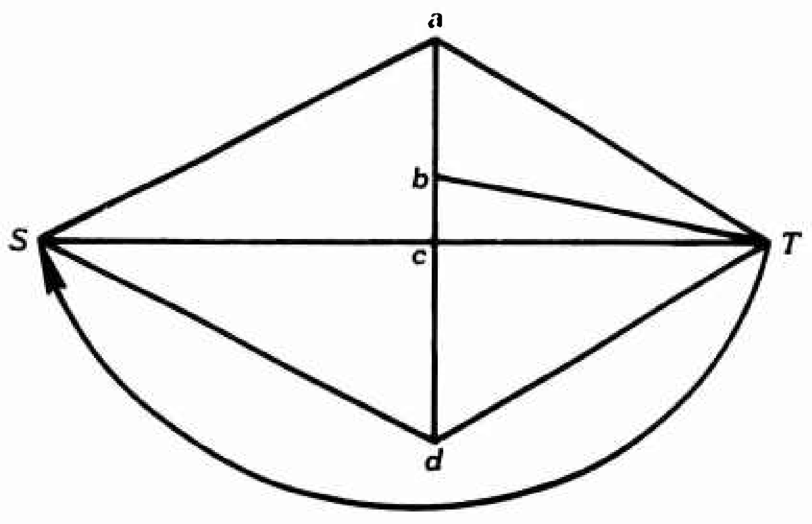

Fig. 5 -- Network of Fig. 1 with artificial arc (T,S) added

Any flow pattern sending x total units of flow from S to T for the network of Fig. 1 can be represented by a circulation flow pattern in the Fig. 5 network by assigning a flow of x units to arc (T,S) and flows on other arcs identical to those of the Fig. 1 network. The reverse is also true.

Network effectiveness to the LOC user, as measured by any of the three criteria of the last subsection, may be found by solving, respectively, the following three problems.

1. Maximize flow from one node (source) to another (sink),
2. Meet a given or required source-sink flow at minimum cost,
3. If it is possible to meet a required source-sink flow, do so at minimum cost. If this is impossible, maximize flow in the least costly way.

These problems may be solved by suitable choices for parameters (other than flow) of the universal arc. The problem of maximizing source-sink flow is converted to one of finding a minimum cost circulation flow by assigning a cost of minus infinity (or a negative number whose absolute value is at least as great as the sum of costs on all other arcs). This insures that all flow cycles incorporating the artificial arc have a negative cost. Hence the minimum cost circulation problem is essentially one of maximizing flow on the universal arc, which is the same

as maximizing source-sink flow. The lower bound on the universal arc would be zero and the upper bound infinity.

A required flow, r, at minimum cost may be met by setting the upper and lower bounds on the universal arc equal to r. Cost on the universal arc, which is arbitrary, would normally be set to zero.

The third problem may be formulated by assigning the universal arc a cost of minus infinity, a lower bound of zero, and an upper bound of the required flow, r. This assures the maximization of flow up to a value of r. Among all flow patterns accomplishing this, the least cost one will be selected. Theoretically, this formulation is also appropriate when it is known that maximum flow can be reduced below required user flow. For this case, however, the strike locations found by the model will normally be closer to optimal if formulated in the context of the first problem.

IV. MODEL OPERATION

DESCRIPTION

From a mathematical viewpoint, it is assumed that the effectiveness of the LOC network to its user may be measured in terms of how cheaply he may achieve a circulation flow. This, of course, includes the physical effectiveness criteria discussed in Sec. III, namely, maximum flow, minimum cost of required flow, and the combined criterion. The model attempts to reduce LOC effectiveness over time, taking into account repair cost, and consequently tries to maximize cost by targeting strikes against the arcs of the LOC network.

Striking an arc decreases its capacity and increases its unit flow cost for a specified period of time, after which its parameters may be restored to their original value by incurring a repair cost. This specified period is referred to as the repair time. In targeting a strike, the arc selected will be the one that maximizes repair cost plus the product of repair time and cost increase of a minimum cost circulation flow. Specifically, if k is the cost of a minimum cost circulation flow before allocating a strike, k_{ij}; this cost after allocating a strike to arc (i,j); t_{ij} and r_{ij}, respectively, the repair time and repair cost for arc (i,j); then the strike will be targeted against the arc that maximizes

$$(1) \qquad (k_{ij}-k)t_{ij} + r_{ij} \ .$$

The above function is quite general and one may derive special cases. For example, repair costs may be eliminated from the calculation by setting them equal to zero. The immediate effect of striking arcs (i.e., not taking time into consideration) may be maximized by setting all repair times equal to one and letting the campaign run for one day.

ARC PARAMETERS AND STRIKES

All strikes allocated against a given arc are directed against a common target.* Each strike has identical but independent probabilities

---

*To see how this restriction may be removed, see Ref. 7.

of success. True arc capacities and unit flow costs are double valued. The first of these values for each are uninterdicted values that apply to an arc when either no strikes are directed against it or some strikes are directed against it but all are unsuccessful; the second are interdicted values that apply when the arc has been struck successfully. Of course, interdicted capacities (unit flow costs) are lower (higher) than uninterdicted ones.

The model expresses arc capacities, unit flow costs, and repair times as functions of the number of strikes, based on expected values. When possible, steady state movement is such that all flow across struck arcs will occur when strikes are unsuccessful or when the arc has been repaired. Thus a targeted arc is either used to its full capacity when uninterdicted or not at all when interdicted. Furthermore, it is assumed that men or equipment delayed in transit, which figure in unit flow costs, can always be used productively for other tasks. Thus no cost penalty is assessed for time spent waiting for arcs to be repaired. For arc (i,j), let u_{ij}^{INT} and u_{ij}^{UNINT} represent interdicted and uninterdicted capacities or upper flow bounds, c_{ij}^{INT} and c_{ij}^{UNINT} interdicted and uninterdicted unit flow costs, r_{ij}^{INT} the true interdicted repair time, and p_{ij} the probability of a strike being successful; and then let u_{ij}^{s}, c_{ij}^{s}, and r_{ij}^{s} represent, as seen by the model, upper flow bound, unit flow cost, and repair cost for arc (i,j) when s strikes are targeted. Then the following relations hold:

$$u_{ij}^{s} = (1-(1-p_{ij})^{s})u_{ij}^{\mathrm{INT}} + (1-p_{ij})^{s}u_{ij}^{\mathrm{UNINT}} \quad , \tag{2}$$

$$c_{ij}^{s} = \begin{cases} c_{ij}^{\mathrm{UNINT}} & \text{for flow units up to } (1-p_{ij})^{s}u_{ij}^{\mathrm{UNINT}} \\ c_{ij}^{\mathrm{INT}} & \text{for flow units beyond } (1-p_{ij})^{s}u_{ij}^{\mathrm{UNINT}} \end{cases} \quad , \tag{3}$$

$$r_{ij}^{s} = (1-(1-p_{ij})^{s})r_{ij}^{\mathrm{INT}} \ . \tag{4}$$

Repair costs in (4) are cumulative. Thus, if arc (i,j) already has been struck s times, the r_{ij} used in (1) would be the difference between r_{ij}^{s+1} and r_{ij}^{s}. Repair time, t_{ij}, is the same for all levels of strike. The quantities k and k_{ij} are both costs of minimum cost circulation flows, but k is the value that results when capacity and unit flow cost for arc (i,j) are u_{ij}^{s} and c_{ij}^{s}, respectively, while k_{ij} results when these quantities are u_{ij}^{s+1} and c_{ij}^{s+1}. Of course, the capacity and unit flow costs for arcs other than (i,j) are the same for the calculation of k_{ij} as they are for k and are those which correspond to the number of strikes already targeted against them.

OPERATION

The model operates in daily cycles, the model user specifying the number of days and strikes per day. At the start of each day, capacities and unit flow costs on arcs scheduled to be returned to their unstruck condition (as specified by their repair times and last time struck) are restored to uninterdicted values. Then the appropriate number of strikes are targeted, one at a time, against the LOC arcs by the second algorithm in Ref. [7]. Each time the arc chosen to be struck is one for which expression (1) is a maximum. Note that this is a marginal allocation procedure, thus giving solutions with the property that if an arc is struck when x number of strikes are targeted, it will also be struck when more than x strikes are targeted. Since optimal placement of strikes would not necessarily have this property, the strike locations are optimal for one strike but only approximate optimality for multiple strikes.

After each strike, information on the arc struck, total source-sink flow, and total cost of this flow is always output. Since the cost on the universal arc is not really incurred, but merely a device to convert a particular type of source-sink flow problem to one of circulation flows, it is not considered in the printed output cost calculation. Costs on artificial arcs are also excluded from this calculation; however, the user may elect to include these costs by inputting them as real arcs. In this regard, the cost given as output differs

from the quantity, k, referred to in expression (1). In particular, targeting an additional strike may result in the output of a lower total cost (and source-sink flow), but could never result in a lower k value. In addition, a detailed status of the network and arc flows is printed at the beginning of each day and at the end of the campaign.

Two other output features are also available at the user's option. First, the status of the entire network may be printed after each strike. Second, a profile of cost versus source-sink flow may be obtained either at the beginning of each day and after the campaign, or at the beginning of each day and after each strike.

When the second option is required, total source-sink flow is divided by the specified number of profile points. Multiples of this number then serve as profile points. The cost of meeting flows is found explicitly at each point. For values of source-sink flow lying between profile points, cost may be approximated by interpolation.

It is important to note that network status and profile points at the beginning of a day may differ from those at the end of the last strike of the previous day. The reason for this is that in the former case any arcs scheduled to return to their unstruck condition are restored to their unstruck values, while in the latter case such restoration has not yet taken place.

Appendix A

PREPARATION OF INPUT DECK

INPUT DATA

Control Data

1. N — is the number of strikes to be targeted each day.
2. PERIOD — is the number of days for which the model is to target strikes and evaluate network status.
3. ICLASS — specifies the classification to be printed on the output. The code is

   1. Unclassified
   2. Confidential
   3. Secret
   4. Top Secret
   5. Confidential, no foreign dissemination
   6. Secret, no foreign dissemination
   7. Top Secret, no foreign dissemination
   8. For Official Use Only

4. IOP — controls the output. If IOP is 1, a paragraph is written on the results of each strike, and a detailed statement of network status is given at the beginning of each day and after the campaign. If IOP is 2, a paragraph on the results of each strike and a detailed status of the network is given after strike as well as at the beginning of each day. If IOP is equal to 3, the information printed is the same as when it is 1, except that the results of a day's strikes are written in tabular form rather than in a series of paragraphs. When the campaign begins, the network status printout consists of information for all arcs so that one has a record of the input network structure; however, subsequent listings contain only arcs with non-zero flow.
5. IART — is the number of artificial arcs in the network (i.e., arcs connecting true sources to the super source and true sinks to the super sink). The universal arc, which

connects the super sink to the super source, is not included in this count.

6. IPROFL determines if and when the profile of flow versus cost is calculated and printed. If 1, the profile option is used at the end of each day; if 2, it is used after each strike; if 3, it is not used at all.

7. NPPTS is the number of profile points (not to exceed 25) desired. Of course, this parameter only applies when IPROFL equals 1 or 2.

8. TOWAY All arcs are considered one-way if this equals 1. If this equals 2, then all arcs with return arcs are considered to be two-way (i.e., both (i,j) and (j,i) exist and are designated by the same arc name or as defined below, ARCNM). Whenever a two-way arc is struck, the strike applies to the return arc as well.

9. JOBS is equal to one.

Arc Data

1. ARCNM is the name of the arc. For two-way arcs, this must be the same for both of the directed arcs. Otherwise, it is used only for identification purposes.
2. FROM is the beginning node of the arc..
3. TO is the ending node of the arc.
4. ICCP is the uninterdicted capacity.
5. LCAP is the interdicted capacity.
6. ICCST is the uninterdicted unit flow cost.
7. UCOST is the interdicted unit flow cost.
8. REPR is the cost of repairing an interdicted or successfully struck arc.
9. ITME is the time required for repairing an interdicted arc.
10, LWER is the lower bound on arc flow.
11. P(I,J) is the probability that a strike is successful.

INPUT FORMAT

A Denotes alphanumeric data.

I Denotes an integer right-adjusted in its field.

D Denotes a decimal fraction. A decimal point must be included in the input.

First Card

Col	1-78	Title of Run	

Second Card

Col	3- 5	Number of Strikes (N)	(I)
Col	8-10	Number of Days (PERIOD)	(I)
Col	13-15	Classification Number (ICLASS)	(I)
Col	18-20	Print Mode (IOP)	(I)
Col	23-25	Number of Artificial Arcs (IART)	(I)
Col	28-30	Profile Option (IPROFL)	(I)
Col	33-35	Number of Profile Points (NPPTS)	(I)
Col	38-40	Two-way Arc Option (TOWAY)	(I)
Col	43-45	Number of Jobs (JOBS)	(I)

Third Card

Col	1- 4	The word "ARCS"	(I)

Arc Data Cards

Each arc is defined by two data cards. The first contains uninterdicted values, the second contains interdicted values. The cards for the universal arc must come first, those for all artificial arcs next, and those for all real arcs last.

First Arc Card

Col	3- 6	Name of Arc (ARCNM)	(A)
Col	10-13	Beginning Node (FROM)	(A)
Col	17-20	Ending Node (TO)	(A)
Col	22-27	Uninterdicted Capacity (ICCP)	(I)
Col	32-37	Uninterdicted Cost (ICCST)	(I)
Col	42-47	Repair Cost (REPR)	(I)
Col	52-57	Repair Time (ITME)	(I)
Col	62-67	Lower Bound (LWER)	(I)
Col	72-75	Kill Probability P(I,J)	(D)

Second Arc Card

Col	3- 6	Name of Arc (ARCNM)	(A)
Col	10-13	Beginning Node (FROM)	(A)
Col	17-20	Ending Node (TO)	(A)
Col	22-27	Interdicted Capacity (LCAP)	(I)
Col	32-37	Interdicted Cost (UCOST)	(I)
Col	42-47	Repair Cost (REPR)	(I)
Col	52-57	Repair Time (ITME)	(I)
Col	62-67	Lower Bound (LWER)	(I)
Col	72-75	Kill Probability P(I,J)	(D)

Next Card

Col	3- 9	The word "COMPUTE"

Appendix B

IBM 360/65 COMPUTER PROGRAM

MAIN

```
C ARCNM ........  THE NAME OF AN ARC.
C AXFLO ........ THE SAME AS MAXFLO(I) BUT IN REAL MODE.
C BSTARC ....... THE SUBSCRIPT OF THE BEST ARC TO STRIKE.
C DAY .......... USED TO COUNT THE CAMPAIGN DAYS,UP TO PERIOD.
C DELTA ........ THE VALUE USED TO LOWER THE UPPER BOUND.
C ERROR ........ A VALUE SENT TO THE ERROR SUBROUTINE SO THAT THE CORRECT
C                MESSAGE IS PRINTED.
C FLOW(I) ...... FLOW ON ARC I.
C FROM(I) ...... 1) AT INPUT TIME CONTAINS THE LOCATION OF THE PREDECESSOR
C                   NODE IN GEOREF OR UTM.
C                2) AFTER NODES ARE NUMBERED IT CONTAINS THE UNIQUE
C                   NODE NUMBER.
C IART ......... THE NUMBER OF ARTIFICIAL ARCS IN THE NETWORK.
C IBSTAC ....... IF THE BEST ARC TO STRIKE WAS TWO-WAY, IT CONTAINS THE
C                SUBSCRIPT OF THE RETURN ARC.
C ICAP(I) ...... UNINTERDICTED CAPACITY FOR ARC I.
C ICCP(I) ...... THE CAPACITY OF ARC I BEFORE ANY INTERDICTION.
C ICCST(I) ..... THE COST ON AN ARC BEFORE ANY INTERDICTION.
C ICLASS ....... CONTROLS THE SECURITY CLASSIFICATION HEADINGS.
C                IF (1) UNCLASSIFIED, (2) CONFIDENTIAL, (3) SECRET, (4)
C                TOP SECRET, (5) CONFIDENTIAL NO FORN, (6) SECRET NO FORN
C                (7) TOP SECRET NO FORN, (8) FOR OFFICIAL USE ONLY.
C ICOST(I) ..... UNINTERDICTED COST FOR ARC I.
C IMNCST ....... THE COST OF THE NETWORK FLOW BEFORE A STRIKE.
C IMNCUT ....... THE COST OF THE NETWORK FLOW AFTER A STRIKE.
C INC .......... THE INCREMENT OF THE PROFILE.
C INCAP(I) ..... CAPACITY OF ARC I INTERDICTED.
C INCOST(I) .... INTERDICTED COST FOR ARC I.
C INCRSE ....... USED IN THE ONECUT SUBROUTINE IN THE CALCULATION
C                OF ARC VALUES.
C INFEAS ....... A SWITCH SET IN MNCF IF THE FLOW PATTERN IS INFEASIBLE.
C IOP .......... AN INPUT WHICH CONTROLS THE OUTPUT, IF IT IS ONE, ONLY
C                THE RESULTS OF A STRIKE ARE PRINTED, IF SET EQUAL TO TWO
C                THE RESULTS OF THE STRIKE AND THE ENTIRE NETWORK ARE
C                PRINTED.  IF EQUAL TO THREE TABULAR OUTPUT IS PRODUCED.
C IPRINT ....... USED IN THE OUTPUT ROUTINE.
C IPROF ........ A SWITCH USED IN THE INNER PRODUCT SUBROUTINE, IF IT
C                IS SET EQUAL TO 0 THE COST OF THE UNIVERSAL ARC IS
C                USED IN THE TOTAL NETWORK COST, IF IT IS SET EQUAL
C                TO 1 THE SUBROUTINE INNER PRODUCT DOES NOT USE THE COST
C                OF THE UNIVERSAL ARC WHEN CALCULATING TOTAL NETWORK
C                COSTS.
C IPROFL ....... IF SET EQUAL TO ONE THE PROFILE SUBROUTINE IS CALLED
C                AT THE END OF EACH DAY. IF = 2 AFTER EACH CUT AND ALSO
C                AT THE END OF EACH DAY.   IF = 3 NOT CALCULATED.
C ISAVCP ....... SAVES THE CAPACITY OF THE RETURN ARC OF BSTARC.
C ISAVE ........ SAVES BSTARC SUBSCRIPT IF IT WAS A TWO-WAY ARC.
C ISVCP1-4 ..... SAVES CAPACITIES IN THE ONECUT SUBROUTINE.
C ISWTCH ....... SAME AS ITWO.
C ITME(I) ...... TIME TO REPAIR ARC I.
C ITWO ......... IF BSTARC WAS ONE-WAY,IT IS EQUAL TO ONE.  IF IT WAS
C                TWO-WAY IT IS SET EQUAL TO TWO.
C JCAP1 ........ SAVES THE CAPACITY OF THE UNINTERDICTED COST PORTION OF
C                BSTARC BEFORE STRIKING.
C JCAP2 ........ SAVES THE ADDITIONAL CAPACITY AT INTERDICTED COST OF
C                BSTARC BEFORE STRIKING.
C JOBS ......... THE NUMBER OF NETWORKS TO BE PROCESSED.
C KCAP1 ........ SAVES THE CAPACITY OF THE UNINTERDICTED COST PORTION OF
C                IBSTAC BEFORE STRIKING.
C KCAP2 ........ SAVES THE ADDITIONAL CAPACITY AT INTERDICTED COST OF
C                IBSTAC BEFORE STRIKING.
C LOWER ........ LOWER BOUND VALUE FOR NETWORK.
C LWER(I) ...... THE LOWER BOUND ON ARC I.
C MAN .......... THE MEN NEEDED FOR REPAIR.
C MAXARC ....... THE NUMBER OF ARCS IN THE NETWORK.
C MAXCST(J) .... THE COST AT THE JTH PROFILE POINT.
C MAXFLO(J) .... THE FLOW AT THE JTH PROFILE POINT.
C MAXI ......... THE SUBSCRIPT OF THE ARC WITH THE GREATEST UPPER BOUND.
C MINCST ....... MINIMUM COST FLOW PATTERN BEFORE THE BEST ARC TO STRIKE IS
C                CHOSEN.
C MINCUT ....... MINIMUM COST FLOW PATTERN WHILE TESTING FOR THE BEST ARC
C                TO STRIKE.
C MXCST ........ A TEMPORARY LOCATION USED TO HOLD THE COST OF A PROFILE.
C N ............ MAXIMUM NUMBER OF STRIKES/DAY.
C NI ........... USED IN THE OUTPUT ROUTINE TO TEST WHICH WRITE TO USE.
C NL(I) ........ USED AS A SCTATCH LIST IN THE MNCF SUBROUTINE.
C NN(I) ........ CONTAINS THE NODE NAMES AFTER THE NODES HAVE BEEN
C                ASSIGNED THEIR UNIQUE NUMBER.
```

```
C NPPTS ....... THE NUMBER OF PROFILE POINTS  DESIRED, CANNOT EXCEED 25.
C NPTS ........ ONE LESS THAN NPPTS,USED IN THE PROFILE SUBROUTINE.
C NR .......... THE NUMBER OF STRIKES PER DAY UP TO N.
C NUM1-4 ...... USED AS TEMPORARY STORAGE IN THE OUTPUT SUBROUTINE.
C PERIOD ...... THE NUMBER OF DAYS THE CAMPAIGN IS TO LAST.
C PK(I) ....... THE PROBABILITY OF KILL FOR ARC I ASSOCIATED WITH THE
C               CHOSEN ATTACK UNIT.
C PPTS ........THE SAME AS NPPTS BUT IN REAL MODE.
C REPAIR(I) ... REPAIR COST FOR ARC I.
C REPR(I) ..... THE REPAIR COST OF AN ARC AFTER INTERDICTION IS DONE.
C RESTOR ...... THE DAY ON WHICH ARC I WILL BE REPAIRED.
C SAVCAP ...... A LOCATION USED TO STORE THE CAPACITY OF ARC MAXI AND
C               BSTARC.
C SAVCST ...... A LOCATION USED TO STORE THE COST OF ARC MAXI AND
C               BSTARC.
C TITLE(4) .... THE ARRAY CONTAINING THE TITLE.
C TO(I) ....... 1) AT INPUT TIME CONTAINS THE LOCATION OF THE SUCCESSOR
C                  NODE IN GEOREF OR UTM.
C               2) AFTER NODES ARE NUMBERED IT CONTAINS THE UNIQUE
C                  NODE NUMBER.
C TOWAY ....... IF SET EQUAL TO 1, ALL ARCS WILL BE CONSIDERED ONE-WAY.
C               IF SET EQUAL TO 2, ARCS WITH RETURN ARCS WILL BE
C               CONSIDERED TWO-WAY, ALL OTHERS WILL BE ONE-WAY.
C UPPER(I) .... THE UPPER BOUND OR CAPACITY OF ARC I.
. XLST ........ AN ARRAY USED AT INPUT TIME CONTAINING SWITCH WORDS.
C
C*******************************************************************
C
C
      COMMON /BLK0/ MINCST,MINCUT,N,INFEAS
      COMMON /BLK1/ IART,PERIOD,DAY,MAXARC,BSTARC,ICLASS,IOP,LINE,S,
C             BSTARC.
     1MAXNDE,JCAP1,JCAP2,KCAP1,KCAP2,NR,IMNCST,IMNCUT,TOWAY
      COMMON /BLK2/ARCNM(800),FROM(800),TO(800),ICOST(800),
     1ICAP(800),INCAP(800),LWER(800),REPR(800),
     2REPAIR(800),ITME(800),FLOW(800),PI(800),NL(800),UPPER(800),
     3IS(800),NN(800),RESTOR(800),ICCP(800),ICCST(800),PK(800)
      COMMON /BLK4/ MAXFLO(25),MAXCST(25),NPPTS,JTEMP(30),TITLE(20)
      COMMON /BLK5/ ITWO,IBSTAC,ISAVCP,ISAVCT,ISWTCH,IPROFL,MAN
      INTEGER PERIOD,DAY,FLOW,FROM,TO,PI,UPPER,BSTARC,S,RESTOR,REPAIR,
     1REPR,ERROR,TOWAY,ARCNM,UCOST
C
C
C
C
C        BRING IN DATA NOW.
C
      CALL INPUT
C
C        NOW NUMBER THE NODES.
C
      CALL NUMBER
C
C        HERE WE WIPE OUT THE NODE NUMBER SCRATCH LIST AND THE FLOW
C        ARRAY.
C
      DO 40 M = 1,MAXARC
      UPPER(M) = 0
      IS(M) = 0
      RESTOR(M) = 0
      NL(M) = 0
      FLOW(M) = 0
   40 CONTINUE
C
C        DESCRIBE THE CAMPAIGN HERE.
C
      CALL OUTPUT(1)
C
C        SET UP A FLOW FOR OUTPUT COST.
C
      CALL MNCF(MAXNDE,MAXARC,FROM,TO,ICOST,ICAP,LWER,FLOW,PI,NL,INFEAS)
      IF(INFEAS .EQ. 1) GO TO 95
C
C        WRITE INITIAL FLOW PATTERN HERE.
C
      DAY=0
      CALL OUTPUT(6)
C
C        HERE WE GET THE COST OF FLOW IN THE NETWORK FOR OUTPUT.
C
```

```
      CALL INPRO(FLOW,ICOST,MAXARC,IMNCST,IART,1)
      CALL OUTPUT(7)
      DAY = 1
      NR = 0
      MAN = 0
C
C        START NEXT STRIKE
C
   70 NR   = NR  + 1
      INFEAS = 0
C
C        IF ASKED FOR, PLOT A PROFILE.
C
      IF(IPROFL .EQ. 2) CALL PROFLE(1)
C
C        GO TO ONECUT TO FIND THE ARC TO STRIKE.
C
      CALL ONECUT
      IF(INFEAS .EQ. 1) GO TO 95
C
C
C        BSTARC IS THE ARC TO STRIKE.
C
C
C        SAVE THE CAPACITY FOR OUTPUT.
C
      KCAP1 = ICAP(IBSTAC)
      KCAP2 = ICAP(IBSTAC + 1)
      JCAP1 = ICAP(BSTARC)
      JCAP2 = ICAP(BSTARC + 1)
      ISAVE = BSTARC
      K = IBSTAC
C
C        WE INCREASE THE NUMBER OF MEN NEEDED FOR REPAIR HERE.
C
      MAN = MAN + REPAIR(K)
   75 CONTINUE
      RESTOR(K) = DAY + ITME(K)
      RESTOR(K+1) = RESTOR(K)
      IS(K) = IS(K) + 1
      IS(K+1) = IS(K)
C
C*********
C
C        ACTUAL INTERDICTION DONE HERE.
C
C*********
C
C
C        UNINTERDICTED COST ARC.
C
      J = IS(K) + 1
      ICAP(K) = INCAP(K)
      INCAP(K) = (1. - PK(K))**J*FLOAT(ICCP(K)) + .5
      REPAIR(K) = FLOAT(REPR(K))*(1. - PK(K))**(J-1)*PK(K) +.5
C
C        INTERDICTED COST ARC.
C
      ICAP(K+1) = INCAP(K+1)
      INCAP(K+1) = (1. -((1. - PK(K)))**J)*FLOAT(ICCP(K+1)) + .5
C
C        IF NOT TWO-WAY SKIP AROUND.
C
      IF(ISWTCH .EQ. 1) GO TO 90
C
C        PRINT THE RESULTS OF STRIKING THE RETURN ARC.
C
      CALL OUTPUT(9)
C
C        NOW GO STRIKE BSTARC.
C
      K = ISAVE
      ISWTCH = 1
      GO TO 75
   90 CONTINUE
C
C        WRITE OUT THE RESULTS OF THE STRIKE HERE.
C
      CALL OUTPUT(2)
C        IF THE FLOW IS LESS THAN THE DEMANDS SAY SO.
C
```

```
   95 CONTINUE
      IF(INFEAS .EQ. 1) GO TO 190
      IF(FLOW(1) .LT. LWER(1)) GO TO 190
      IF(FLOW(1) .EQ. 0 ) GO TO 200
      IF(BSTARC .EQ. 1) GO TO 100
C
C        CHECK TO SEE IF DAY IS DONE
C
      IF(NR .EQ. N    ) GO TO 100
      GO TO 70
C
C        CHECK TO SEE IF CAMPAIGN IS DONE
C
  100 IF(DAY .EQ. PERIOD    ) GO TO 210
C
C        IF ASKED FOR, PLOT A PROFILE.
C
      IF(IPROFL .NE. 3) GO TO 109
      IF(IOP .EQ. 2) LINE = 0
      IF(IOP .NE. 2) LINE = 48
  109 IF(IPROFL .LE. 2) CALL PROFLE(1)
C
C        UPDATE FOR NEXT DAY
C
  110 DAY = DAY + 1
      NR= 0
C
C        IF AN ARC IS TO BE REPAIRED, RESTORE ALL ZERO STRIKE VALUES.
C
      DO 160 M = S,MAXARC,2
      IF(RESTOR(M) .GT. DAY) GO TO 160
      IF(IS(M) .EQ. 0) GO TO 160
C
C        INITIALIZE THE UNINTERDICTED COST ARC.
C
      ICAP(M) = ICCP(M)
      INCAP(M) = (1. - PK(M))*FLOAT(ICCP(M)) + .5
      REPAIR(M) = PK(M) * FLOAT(REPR(M)) + .5
      IS(M) = 0
      RESTOR(M) = 0
  160 CONTINUE
      MM = S + 1
      DO 170 M = MM,MAXARC,2
C
C        INITIALIZE THE INTERDICTED COST ARC.
C
      IF(RESTOR(M) .GT. DAY) GO TO 170
      IF(IS(M) .EQ. 0) GO TO 170
      ICAP(M) = 0
      INCAP(M) = (1. -(1. -PK(M-1)))*FLOAT(ICCP (M)) + .5
      IS(M) = 0
      RESTOR(M) = 0
  170 CONTINUE
C
C        SET UP FLOW FOR THE NEXT DAY.
C
      CALL MNCF(MAXNDE,MAXARC,FROM,TO,ICOST,ICAP,LWER,FLOW,PI,NL,INFEAS)
      IF(INFEAS .EQ. 1) GO TO 95
C
C        GET ITS COST.
C
      CALL INPRD(FLOW,ICOST,MAXARC,IMNCST,IART,1)
C
C        PRINT NETWORK FLOW PATTERN AND ITS COST.
C
      CALL OUTPUT(6)
      CALL OUTPUT(7)
      GO TO 70
  190 CALL OUTPUT(3)
      GO TO 100
  200 CALL OUTPUT(4)
      GO TO 100
  210 IF (IOP .NE. 2) LINE = 48
      IF (IOP .NE. 2) CALL OUTPUT (6)
      CALL OUTPUT(7)
C
C        IF ASKED FOR, PLOT A PROFILE.
C
      IF(IPROFL .LE. 2) CALL PROFLE(1)
      CALL EXIT
      END
```

INPRO

```
      SUBROUTINE INPRO(FLOW,COST,MAXARC,MINCT,IART,IPROF)
      DIMENSION FLOW(1),COST(1)
      INTEGER FLOW   , COST
C
C
C
      MINCT  = 0
      ITRT = 2 * IART + 1
C
C        IF IPROF IS EQUAL TO 0 THEN USE THE COSTS ON THE UNIVERSAL ARC.
C
      IF(IPROF .EQ. 0) ITRT = 1
      DO 10 I = ITRT,MAXARC
   10 MINCT = MINCT + FLOW(I)*COST(I)
      RETURN
      END
```

INPUT

```
      SUBROUTINE  INPUT
      DIMENSION            TEMP(20),XLST(2)
      COMMON /BLK0/ MINCST,MINCUT,N,INFEAS
      COMMON /BLK1/ IART,PERIOD,DAY,MAXARC,BSTARC,ICLASS,IOP,LINF,S,
     1MAXNDE,JCAP1,JCAP2,KCAP1,KCAP2,NR,IMNCST,IMNCUT,TOWAY
      COMMON /BLK2/ARCNM(800),FROM(800),TO(800),ICOST(800),
     1ICAP(800),INCAP(800),LWER(800),REPR(800),
     2REPAIR(800),ITME(800),FLOW(800),PI(800),NL(800),UPPER(800),
     3IS(800),NN(800),RESTOR(800),ICCP(800),ICCST(800),PK(800)
      COMMON /BLK4/ MAXFLO(25),MAXCST(25),NPPTS,JTEMP(30),TITLE(20)
      COMMON /BLK5/ ITWC,IBSTAC,ISAVCP,ISAVCT,ISWTCH,IPROFL,MAN
      INTEGER PERIOD,DAY,FLOW,FROM,TO,PI,UPPER,BSTARC,S,RESTOR,REPAIR,
     1REPR,ERROR,TOWAY,ARCNM,UCOST
      INTEGER TEMP,XLST
      DATA XLST/'ARCS','COMP'/
C
C
C
      I=0
C
C        READ THE TITLE CARD AND SKIP A PAGE.
C
      READ( 5 ,905)(TITLE(MM),MM=1,20)
      READ( 5 ,900)                  N,PERIOD,ICLASS,IOP,IART
     1,IPROFL,NPPTS,TOWAY,JOBS
      WRITE( 6 ,904)
      S = 2 * IART + 1
C
C        LOOK FOR THE ARCS CARD.
C
      READ( 5 ,905)(TEMP(J),J=1,20)
      IF(TEMP(1) .EQ.XLST(1)) GO TO 10
C
C        IF NOT THE ARCS CARD, ERROR.
C
      CALL EXIT
C
C        READ IN THE ARC DATA
C
   10 I=I+1
C
C        READ AN UNINTERDICTED COST ARC.
C
      READ( 5 ,910)ARCNM(I),FROM(I),TO(I),ICCP(I),ICCST(I),REPR(I),
     1ITME(I),LWER(I),PK(I)
      IF(ARCNM(I) .EQ.XLST(2))  GO  TO  30
C
C        INITIALIZE THE UNINTERDICTED COST ARC.
C
      ICAP(I) = ICCP(I)
      ICOST(I) = ICCST(I)
      INCAP(I) = (1. - PK(I))*FLOAT(ICCP(I)) + .5
      REPAIR(I) = PK(I) * FLOAT(REPR(I)) + .5
      I = I + 1
C
C        READ THE CORRESPONDING INTERDICTED COST ARC.
C
      READ( 5 ,910)ARCNM(I),FROM(I),TO(I),ICCP(I),ICCST(I),REPR(I),
     1ITME(I),LWER(I),PK(I)
C
C        INITIALIZE THE INTERDICTED COST ARC.
C
      ICAP(I) = 0
      ICOST(I) = ICCST(I)
      INCAP(I) = (1. -(1. -PK(I-1)))*FLOAT(ICCP (I)) + .5
C
      GO TO 10
   30 CONTINUE
      MAXARC = I - 1
      RETURN
C
C
C
  900 FORMAT(   16(2X,I3))
  904 FORMAT(1H1,10X)
  905 FORMAT(20A4)
  910 FORMAT(2X,2(A4,3X),A4,1X,5(I6,4X),F5.4)
      END
```

MNCF

```
      SUBROUTINE MNCF(NODES,ARCS,I,J,COST,HI,LO,FLOW,PI,NA,INFEAS)
C     MNCF IS THE FULKERSON OUT OF KILTER ALGORITHM.
      INTEGER NODES,ARCS,I,J,COST,HI,LO,FLOW,PI,NA,INFEAS
      DIMENSION I(3000),J(3000),COST(3000),HI(3000),LO(3000),FLOW(3000)
      DIMENSION PI(1000),NA(1000)
C
C
C     DEFINITION OF CALLING SEQUENCE
C
C
C    NAME    USE
C
C    NODES   NUMBER OF NODES
C    ARCS    NUMBER OF ARCS
C    I       LIST OF FROM ZBEGINNINGC NODES
C    J       LIST OF TO ZENDINGC NODES
C    COST    UNIT COST OF FLOW ON ARCS
C    HI      UPPER BOUNDS FOR ARCS
C    LO      LOWER BOUNDS FOR ARCS
C    FLOW    AMOUNT OF FLOW IN ARCS
C    PI      NODE PRICES
C    NA      SCRATCH LIST FOR NODE LABELLING
C    INFEAS  FLAG DENOTING THE CONDITION OF OUTPUT
C
C   BEGIN
      INTEGER A,AA,N,SRC,SNK,DEL,INF,C,AOK,COK,N1,N2,INC,LABEL
      INF = 2147483647
      AOK = 0
C LOOK FOR AN OUT OF KILTER ARC
C     DO 90 AA=1, ARCS
      AA = 1
  100   N1 =  I(AA)
        N2 =  J(AA)
        C  =  COST(AA) +PI(N1) - PI(N2)
   40   IF(FLOW(AA).LT.LO(AA).OR.(C.LT.0.AND.FLOW(AA).LT.HI(AA)))GOTO
        IF(FLOW(AA).GT.HI(AA).OR.(C.GT.0.AND.FLOW(AA).GT.LO(AA)))GOTO
C   90 CONTINUE
   90 AA = AA + 1
      IF(AA.LE.ARCS)GO TO 100
C   NO OUT OF KILTER ARCS LEFT
      INFEAS = 0
      RETURN
C     OUT OF KILTER ARC FOUND
   50   SRC = J(AA)
        SNK = I(AA)
      LABEL = +AA
        GO TO 200
   60   SRC = I(AA)
        SNK = J(AA)
      LABEL = -AA
C   SAVE LABELS IF LAST OPERATION WAS INCREASING NODE PRICES ON THIS ARC
  200 IF ( AA.EQ.AOK.AND.NA(SRC).NE.0)  GO TO 205
      DO 201 N = 1,NODES
        NA(N) = 0
  201 CONTINUE
      AOK =AA
  205 COK = C
      NA(SRC) = LABEL
C LABEL
  210 LABEL = 0
      DO 250 A =1,ARCS
        N1 = I(A)
        IF (N1.LT.0) GO TO 250
        N2 = J(A)
        IF  (NA(N1).EQ.0.AND.NA(N2).EQ.0) GO TO 250
        IF (NA(N1).NE.0.AND.NA(N2).NE.0)  GO TO 245
        C = COST(A) + PI(N1) - PI(N2)
        IF (NA(N1).EQ.0)  GO TO 220
        IF(FLOW(A).GE.HI(A).OR.(FLOW(A).GE.LO(A).AND.C.GT.0)) GO TO 245
        NA(N2) = A
        GO TO 240
  220   IF(FLOW(A).LE.LO(A).OR.(FLOW(A).LE.HI(A).AND.C.LT.0)) GO TO 245
        NA(N1) = -A
  240   LABEL = 1
C   NODE LABELED, TEST FOR BREAKTHRU
        IF (NA(SNK).NE.0) GO TO 260
  245   I(A) = -N1
  250 CONTINUE
```

```
C  GO BACK AND DO MORE LABELING IF SOME NODE WAS LABELED ON LAST PASS
      IF (LABEL.NE.0)  GO TO 210
C  RESTORE POSITIVE SIGNS TO FIRST NODE LIST
  260 DO 270 A = 1,ARCS
        I(A) = IABS(I(A))
  270 CONTINUE
C  IF NO LABELING DONE ON LAST PASS, GO TO INCEASE PIE
      IF (LABEL.EQ.0) GO TO 400
C  BREAKTHRU,  FIND THE INCREMENT
  300 INC = INF
C  FOLLOW PATH BACK FROM SOURCE
      N    = SNC
  310 A = IABS(NA(N))
      IF (NA(N).LT.0)  GO TO 315
      N2 = I(A)
      C = COST(A) - PI(N) + PI(N2)
      IF (C.GT.0) INC = MIN0(INC,LO(A)-FLOW(A) )
      IF (C.LE.0) INC = MIN0(INC,HI(A)-FLOW(A) )
      GO TO 340
  315 N2 = J(A)
      C = COST(A) + PI(N) - PI(N2)
  320 IF (C.LT.0) INC = MIN0(INC,FLOW(A)-HI(A) )
      IF (C.GE.0) INC = MIN0(INC,FLOW(A)-LO(A) )
  340 N   = N2
      IF  (N.NE.SRC) GO TO 310
C  INCREMENT ARCS
  350 A = IABS(NA(N))
      IF (NA(N).LT.0)  GO TO 360
      FLOW(A) = FLOW(A) + INC
      N = I(A)
      GO TO 370
  360 FLOW(A) = FLOW(A) - INC
      N = J(A)
  370 IF ( N.NE.SRC ) GO TO 350
C  FLOW INCREMENTED, RETURN TO KILTER TEST
      NA(N) = 0
      GO TO 100
C  CHANGE PI
  400 DEL = INF
C  FIND INCREMENT
      DO 420 A=1,ARCS
        N1 = I(A)
        N2 = J(A)
        IF    (NA(N1).EQ.0 .AND. NA(N2).EQ.0)  GO TO 420
        IF    (NA(N1).NE.0 .AND. NA(N2).NE.0)  GO TO 420
        C   = COST(A) + PI(N1) - PI(N2)
        IF (NA(N2).EQ.0 .AND. FLOW(A).LT.HI(A) ) DEL=MIN0(DEL,C)
        IF (NA(N2).NE.0 .AND. FLOW(A).GT.LO(A) ) DEL=MIN0(DEL,-C)
  420 CONTINUE
      IF (DEL.NE.INF)  GO TO 430
      IF  (FLOW(AA).EQ.LO(AA)  .OR.  FLOW(AA).EQ.HI(AA))  GO TO 425
C  INFEASIBLE SOLUTION
      INFEAS = 1
      RETURN
C  INCREASE PI
  425 DEL = IABS(COK)
  430 DO 450 N =1,NODES
        IF (NA(N).EQ.0)  PI(N) = PI(N) + DEL
  450 CONTINUE
C  GO BACK TO KILTER TEST
      GO TO 100
      END
```

NODENO

```
      FUNCTION NODENO(III,NN)
      DIMENSION NN(1)
      COMMON /BLK1/ IART,PERIOD,DAY,MAXARC,BSTARC,ICLASS,IOP,LINE,S,
     1MAXNDE,JCAP1,JCAP2,KCAP1,KCAP2,NR,IMNCST,IMNCUT,TOWAY
C
C
C
      NODENO = MAXNDE + 1
      IF(MAXNDE .EQ. 0) RETURN
      DO 10 I = 1, MAXNDE
      IF(III .EQ. NN(I)) NODENO=I
   10 CONTINUE
      RETURN
      END
```

NUMBER

```
      SUBROUTINE NUMBER
      COMMON /BLK0/ MINCST,MINCUT,N,INFEAS
      COMMON /BLK1/ IART,PERIOD,DAY,MAXARC,BSTARC,ICLASS,IOP,LINE,S,
     1MAXNDE,JCAP1,JCAP2,KCAP1,KCAP2,NR,IMNCST,IMNCUT,TOWAY
      COMMON /BLK2/ARCNM(800),FROM(800),TO(800),ICOST(800),
     1ICAP(800),INCAP(800),LWER(800),REPR(800),
     2REPAIR(800),ITME(800),FLOW(800),PI(800),NL(800),UPPER(800),
     3IS(800),NN(800),RESTOR(800),ICCP(800),ICCST(800),PK(800)
      COMMON /BLK5/ ITWO,IBSTAC,ISAVCP,ISAVCT,ISWTCH,IPROFL,MAN
      INTEGER PERIOD,DAY,FLOW,FROM,TO,PI,UPPER,BSTARC,S,RESTOR,REPAIR,
     1REPR,ERROR,TOWAY,ARCNM,UCOST
C
C
C
C
C        NUMBER THE NODES
C
      MAXNDE = 0
      DO 30 L=1,MAXARC
      NN(MAXNDE+1) =FROM(L)
      FROM(L) =NODENO(FROM(L),NN)
      MAXNDE = MAX0(MAXNDE,FROM(L))
   30 CONTINUE
      DO 40 L = 1, MAXARC
      NN(MAXNDE+1) = TO(L)
      TO(L) = NODENO(TO(L),NN)
      MAXNDE = MAX0(MAXNDE,TO(L))
   40 CONTINUE
C
C        CHECK FOR DEAD NODES
C
      DO 48 I = 1,MAXNDE
      DO 42 L = 1,MAXARC
      IF(FROM(L) .EQ. I) GO TO 44
   42 CONTINUE
      WRITE( 6 ,925) NN(I)
   44 CONTINUE
      DO 46 L = 1,MAXARC
      IF(TO(L) .EQ. I ) GO TO 48
   46 CONTINUE
      WRITE( 6 ,930) NN(I)
   48 CONTINUE
      RETURN
C
C
C
  925 FORMAT(1H1, 22HNO ARC BEGINS AT NODE , A4/1H1)
  930 FORMAT(1H1, 22HNO ARC ENDS AT NODE   , A4/1H1)
      END
```

ONECUT

```
      SUBROUTINE ONECUT
      COMMON /BLK0/ MINCST,MINCUT,N,INFEAS
      COMMON /BLK1/ IART,PERIOD,DAY,MAXARC,BSTARC,ICLASS,IOP,LINE,S,
     1MAXNDE,JCAP1,JCAP2,KCAP1,KCAP2,NR,IMNCST,IMNCUT,TOWAY
      COMMON /BLK2/ARCNM(800),FROM(800),TO(800),ICOST(800),
     1ICAP(800),INCAP(800),LWER(800),REPR(800),
     2REPAIR(800),ITME(800),FLOW(800),PI(800),NL(800),UPPER(800),
     3IS(800),NN(800),RESTOR(800),ICCP(800),ICCST(800),PK(800)
      COMMON /BLK4/ MAXFLO(25),MAXCST(25),NPPTS,JTEMP(30),TITLE(20)
      COMMON /BLK5/ ITWO,IBSTAC,ISAVCP,ISAVCT,ISWTCH,IPROFL,MAN
      INTEGER PERIOD,DAY,FLOW,FROM,TO,PI,UPPER,BSTARC,S,RESTOR,REPAIR,
     1REPR,ERROR,TOWAY,ARCNM,UCOST
      INTEGER DELTA,DECRSE
C
C
C
C         HERE WE INITIALIZE.
C         ARCS FOR WHICH THE NUMBER OF STRIKES IS THE SAME AS WHEN THIS
C         ROUTINE IS ENTERED WILL BE CONSIDERED UNSTRUCK.  ONE TO WHICH
C         AN ADDITIONAL STRIKE HAS BEEN ADDED WILL BE CONSIDERED STRUCK.
C
      MINCUT = 0
      MINCST = 0
      BSTARC = 1
      LOWER = 0
C
C         SET UPPER BOUNDS AT INFINITY.
C
      DO 10 I = S,MAXARC,2
   10 UPPER(I) = 2147483647
C
C         START FLOW BEFORE WE SELECT AN ARC.
C
      CALL MNCI(MAXNDE,MAXARC,FROM,TO,ICOST,ICAP,LWER,FLOW,PI,NL,INFEAS)
C
C         IF INFEASIBLE FLOW RETURN.
C
      IF( INFEAS .EQ. 1 ) RETURN
C
C         GET COST OF FLOW IN THE SYSTEM.
C
      CALL INPROD(FLOW,ICOST,MAXARC,MINCST,IART,0)
C
C         IF THE FLOW IS LESS THAN THE STRUCK CAPACITY LOWER THE
C         UPPER BOUND, OTHERWISE THE UPPER BOUND DOES NOT CHANGE.
C
      DO 30 I = S,MAXARC,2
      IF(FLOW(I) + FLOW(I+1) .GT. INCAP(I) + INCAP(I+1)) GO TO 30
      INCRSE = MAX0(0,FLOW(I) - INCAP(I))*(ICOST(I+1)-ICOST(I))
      UPPER(I) = INCRSE * ITME(I) + REPAIR(I)
   30 CONTINUE
   35 CONTINUE
C
C         FIND THE ARC WITH THE GREATEST UPPER BOUND, IT IS MAXI.
C
      MAXI = 1
      DO 50 I = S,MAXARC,2
      IF(UPPER(I) .GT. UPPER(MAXI)) GO TO 40
      GO TO 50
   40 MAXI = I
   50 CONTINUE
C
C         IF THE GREATEST UPPER BOUND IS EQUAL TO THE LOWER BOUND WE ARE
C         DONE, THE ARC TO STRIKE IS BSTARC.
C
      IF(UPPER(MAXI).NE.LOWER) GO TO 60
C
C         TEST TO SEE IF BSTARC WAS TWO-WAY.
C
      DO 52 I =S,MAXARC,2
      IF(FROM(BSTARC) .EQ. TO(I) .AND. TO(BSTARC) .EQ. FROM(I)
     1.AND. ARCNM(BSTARC) .EQ. ARCNM(I)) GO TO 54
   52 CONTINUE
   53 CONTINUE
C
C         BSTARC IS NOT TWO-WAY.
C
```

```
      ITWO = 1
      IBSTAC = BSTARC
      GO TO 56
C
C        BSTARC WAS TWO-WAY.
C
   54 ITWO = 2
      IBSTAC = I
      IF(TOWAY .EQ. 1) GO TO 53
C
C        NOW STORE AWAY THE CAPACITY OF THE BEST ARC AND ITS BYPASS.
C
   56 CONTINUE
      ISVCP1 = ICAP(IBSTAC)
      ISVCP2 = ICAP(IBSTAC + 1)
      ISVCP3 = ICAP(BSTARC)
      ISVCP4 = ICAP(BSTARC + 1)
      ISWTCH = ITWO
C
C        SET THE UNINTERDICTED AND INTERDICTED COST CAPACITIES TO
C        THEIR STRUCK VALUES.
C
      ICAP(IBSTAC) = INCAP(IBSTAC)
      ICAP(BSTARC) = INCAP(BSTARC)
      ICAP(IBSTAC + 1) = INCAP(IBSTAC + 1)
      ICAP(BSTARC + 1) = INCAP(BSTARC + 1)
C
C        SET UP THE FLOW PATTERN WITH THE BEST ARC INTERDICTED.
C
      CALL MNCF(MAXNDE,MAXARC,FROM,TO,ICOST,ICAP,LWER,FLOW,PI,NL,INFEAS)
C
C        IF INFEASIBLE FLOW RETURN.
C
      IF(INFEAS .EQ. 1 ) RETURN
C
C        HERE WE GET THE COST OF FLOW IN THE NETWORK FOR OUTPUT.
C
      CALL INPRD(FLOW,ICOST,MAXARC,IMNCUT,IART,1)
C
C        RESTORE THE CAPACITY OF THE BEST ARC AND ITS BYPASS SO THAT THE
C        INTERDICTION CAN BE DONE AT ONE PLACE IN THE MAIN PROGRAM.
C
      ICAP(IBSTAC) = ISVCP1
      ICAP(IBSTAC + 1) = ISVCP2
      ICAP(BSTARC) = ISVCP3
      ICAP(BSTARC + 1) = ISVCP4
C
C        GET THE VALUE OF STRIKING THE BEST ARC THEN RETURN.
C
      MINCUT = MINCST + (LOWER - REPAIR(BSTARC))/ITME(BSTARC)
      RETURN
C
C        SINCE WE DO NOT HAVE THE BEST ARC TO STRIKE WE SAVE THE
C        UNINTERDICTED AND INTERDICTED COST CAPACITIES OF THE ARC
C        WITH THE GREATEST UPPER BOUND.
C
   60 SAVCAP = ICAP(MAXI)
      ISAVCP = ICAP(MAXI + 1)
C
C        WE NOW SEE WHAT WOULD BE THE COST IF THE ARC WITH THE GREATEST
C        UPPER BOUND IS SELECTED.
C
      ICAP(MAXI) = INCAP(MAXI)
      ICAP(MAXI + 1) = INCAP(MAXI + 1)
C
C        SET UP THE FLOW WITH MAXI OUT.
C
      CALL MNCF(MAXNDE,MAXARC,FROM,TO,ICOST,ICAP,LWER,FLOW,PI,NL,INFEAS)
C
C        IF INFEASIBLE FLOW RETURN.
C
      IF(INFEAS .EQ. 1 ) RETURN
C
C        NOW GET ITS COST.
C
      CALL INPRD(FLOW,ICOST,MAXARC,MINCUT,IART,0)
C
C        CALCULATE A NEW UPPER BOUND.
C
      UPPER(MAXI) =(MINCUT - MINCST)*ITME(MAXI) + REPAIR(MAXI)
```

```
C
C        IF THE UPPER BOUND IS STILL GREATER THAN THE LOWER BOUND,
C        MAXI BECOMES THE BEST ARC TO STRIKE.
C
      IF(UPPER(MAXI) .GT. LOWER) GO TO 70
      GO TO 80
   70 LOWER = UPPER(MAXI)
      BSTARC = MAXI
   80 CONTINUE
      DO 100 I = 5,MAXARC,2
C
C        IF THE UPPER BOUND ON AN ARC IS LESS THAN THE LOWER BOUND OR
C        NO CHANGE ON THE UPPER BOUND.
C
      IF(UPPER(I) .LE. LOWER) GO TO 100
C
C        IF THE FLOW IS GREATER THAN THE STRUCK CAPACITY ALSO
C        NO CHANGE.
C
      IF(FLOW(I) + FLOW(I+ 1) .GT. INCAP(I) + INCAP(I + 1)) GO TO 100
C
C        OTHERWISE CALCULATE DELTA.
C
      INCRSE = MAX0(0,FLOW(I) -INCAP(I))*(ICOST(I + 1) - ICOST(I))
      DECRSE = MIN0(ICAP(MAXI) - INCAP(MAXI),FLOW(MAXI + 1))*
     1(ICOST(MAXI + 1) - ICOST(MAXI))
      DELTA = (MINCUT + INCRSE - DECRSE - MINCST) * ITME(I) + REPAIR(I)
C
C        IF DELTA IS LESS THAN THE UPPER BOUND, IT BECOMES THE NEW
C        UPPER BOUND, OTHERWISE NO CHANGE.
C
      IF(DELTA .GE. UPPER(I)) GO TO 100
      UPPER(I) = DELTA
  100 CONTINUE
C
C        RESTORE THE CAPACITY AND GO GET ANOTHER ARC.
C
      ICAP(MAXI) = SAVCAP
      ICAP(MAXI+1) = ISAVCP
      GO TO 35
      END
```

OUTPUT

```
      SUBROUTINE OUTPUT(J)
      COMMON /BLK0/ MINCST,MINCUT,N,INFEAS
      COMMON /BLK1/ IANT,PERIOD,DAY,MAXARC,BSTARC,ICLASS,IOP,LINE,S,
     1MAXNDE,JCAP1,JCAP2,KCAP1,KCAP2,NR,IMNCST,IMNCUT,TOWAY
      COMMON /BLK2/ARCNM(800),FROM(800),TO(800),ICOST(800),
     1ICAP(800),INCAP(800),LWER(800),REPR(800),
     2REPAIR(800),ITME(800),FLOW(800),PI(800),NL(800),UPPER(800),
     3IS(800),NN(800),RESTOR(800),ICCP(800),ICCST(800),PK(800)
      COMMON /BLK4/ MAXFLO(25),MAXCST(25),NPPTS,JTEMP(30),TITLE(20)
      COMMON /BLK5/ ITWO,IBSTAC,ISAVCP,ISAVCT,ISWTCH,IPROFL,MAN
      INTEGER PERIOD,DAY,FLOW,FROM,TO,PI,UPPER,BSTARC,S,RESTOR,REPAIR,
     1REPR,ERROR,TOWAY,ARCNM,UCOST
C
C
C
      IF (J .GT. 1 ) GO TO 180
      CALL PAGE(6,ICLASS)
      WRITE( 6 ,906)(TITLE(I),I=1,20)
      WRITE( 6 ,907)PERIOD,N
      CALL PAGE(6,ICLASS)
      WRITE( 6 ,904)
      LINE = 0
      RETURN
   20 IARC = 0
   30 LINE = 0
      WRITE( 6 ,908)
   40 IARC = IARC + 2
      IF(IARC .GT. MAXARC  ) GO TO 50
      IF (FLOW(IARC-1) .EQ. 0 .AND. DAY .NE. 0) GO TO 40
      IF(FLOW(IARC-1) .EQ. 0 .AND. J .NE. 6) GO TO 40
      LINE = LINE + 1
      II = FROM(IARC)
      IJ = TO(IARC)
      WRITE( 6 ,909)ARCNM(IARC-1),NN(II),NN(IJ),ICAP(IARC-1),ICOST(IARC-
     11),REPAIR(IARC-1),ITME(IARC-1),LWER(IARC-1),FLOW(IARC-1),RESTOR(IA
     2RC-1),IS(IARC-1),ICAP(IARC),ICOST(IARC),FLOW(IARC),PK(IARC-1)
      IF(LINE .NE. 49) GO TO 40
      CALL PAGE(6,ICLASS)
      WRITE( 6 ,904)
      CALL PAGE(6,ICLASS)
      GO TO 30
   50 CALL PAGE(6,ICLASS)
      WRITE( 6 ,904)
      LINE = 0
      RETURN
   60 CONTINUE
      IF(LINE + 6 .GE. 48) GO TO 430
      IF(IPROFL .EQ. 2) GO TO 65
      IF(LINE .NE. 0) GO TO 70
   65 CONTINUE
      WRITE( 6 ,900)
      LINE = LINE + 4
   70 CONTINUE
      IPRINT = BSTARC
      IF(ISWTCH .EQ. 2) IPRINT = IBSTAC
      WRITE( 6 ,901) DAY,NR,ARCNM(IPRINT),NN(NUM1),NN(NUM2),
     1RESTOR(IPRINT),NUM3,NUM4,JCAP1,ICAP(IPRINT),MAN,IMNCUT,IS(IPRINT),
     2FLOW(1)
      LINE = LINE + 1
      RETURN
  180 IF(LINE .NE. 48)GO TO 230
      CALL PAGE(6,ICLASS)
      WRITE( 6 ,904)
      LINE = 0
      IF(J .EQ. 5) RETURN
  230 IF(LINE .NE. 0 ) GO TO 280
      CALL PAGE(6,ICLASS)
  280 GO TO(320,320,330,340,320,350,360,370,420),J
  320 CONTINUE
      NUM1 =  FROM(BSTARC)
      NUM2 =  TO(BSTARC)
      NUM3 = JCAP1 + JCAP2
      NUM4 = ICAP(BSTARC) + ICAP(BSTARC + 1)
      IF(IOP .EQ. 3) GO TO 60
      IF(LINE + 8 .GE. 48) GO TO 430
      WRITE( 6 ,910) DAY,NR,ARCNM(BSTARC),NN(NUM1),NN(NUM2),
     1RESTOR(BSTARC),NUM3,JCAP1,NUM4,ICAP(BSTARC),MAN,IMNCUT,
     2IS(BSTARC),FLOW(1)
```

```
      LINE = LINE + 8
      IF(IOP .EQ. 1) RETURN
      CALL PAGE(6,ICLASS)
      WRITE( 6 ,904)
      LINE = 0
      CALL PAGE(6,ICLASS)
      GO TO 20
  330 WRITE( 6 ,913) DAY
      LINE = LINE + 8
      RETURN
  340 WRITE( 6 ,912) DAY
      LINE = LINE + 8
      RETURN
  350 GO TO 20
  360 WRITE( 6 ,915) IMNCST,FLOW(1)
      CALL PAGE(6,ICLASS)
      WRITE( 6 ,904)
      LINE = 0
      RETURN
  370 CONTINUE
      IF(LINE+16 .GE. 49) GO TO 430
      WRITE( 6 ,916)
      IF(NPPTS .GT. 12) GO TO 390
      DO 380 I = 1,NPPTS
      WRITE( 6 ,917) I,MAXFLO(I),MAXCST(I)
  380 CONTINUE
  385 GO TO 50
  390 I = 0
  400 I = I + 1
      IF(I .GT. 12) GO TO 385
      NI = I + 12
      IF(NI .GT. NPPTS) GO TO 410
      WRITE( 6 ,918) I,MAXFLO(I),MAXCST(I),NI,MAXFLO(NI),MAXCST(NI)
      GO TO 400
  410 WRITE( 6 ,917) I,MAXFLO(I),MAXCST(I)
      GO TO 400
  420 CONTINUE
      NUM1 =  TO(BSTARC)
      NUM2 =  FROM(BSTARC)
      NUM3 = KCAP1 + KCAP2
      NUM4 = ICAP(IBSTAC) + ICAP(IBSTAC + 1)
      IF(IOP .EQ. 3) GO TO 60
      IF(LINE + 8 .GE. 48) GO TO 430
      WRITE( 6 ,910) DAY,NR,ARCNM(IBSTAC),NN(NUM1),NN(NUM2),
     1RESTOR(IBSTAC),NUM3,KCAP1,NUM4,ICAP(IBSTAC),MAN,IMNCUT,
     2IS(IBSTAC),FLOW(1)
      LINE = LINE + 8
      RETURN
  430 LINE = 48
      GO TO 180
C
C
C
  900 FORMAT(1H ,/23X,15HSTRUCK  STRUCK ,11X,14HTOTAL EXPECTED,4X,
     118HMAX. NO. UNITS ARC,15X,25HEXPECTED  NUMBER    TOTAL/
     2  17X,46HNAME      ARC      ARC    DAY ARC   ARC CAPACITY,5X,
     3  59HFLOW AT UNINT. COST    TOTAL     NETWORK    TIMES  EXPECTED/
     429H  DAY  MISSION  OF ARC   FROM,   5X,2HTO,6X,14HTO BE   BEFORE,
     53X, 5HAFTER,6X,14HBEFORE   AFTER,6X,26HREPAIR    FLOW COST    ARC,
     610H   NETWORK,
     7/58H  NO.    NO.    STRUCK   NODE     NODE   RESTORED  STRIKE  ,
     8 59HSTRIKE       STRIKE  STRIKE       COST       PER DAY   STRUCK,
     9 8H    FLOW)
  901 FORMAT(2H  ,I3,4X,I3,1X,3(4X,A4),6X,I3,2X,2(3X,I5),4X,2(3X,I5),
     12X,2(3X,I8),5X,I3,3X,I8)
  904 FORMAT(1H1,10X)
  905 FORMAT( /1H ,10X)
  906 FORMAT(1H , 24X,20A4 ///)
  907 FORMAT(1H ,32X, 23HCAMPAIGN WILL LAST FOR   ,I3,12H DAYS, WITH  ,
     1I3,20H MISSIONS PER DAY.         )
  908 FORMAT(1H ,24X,15HCAP. AT  UNINT.,18X,23HLOWER  FLOW AT    TO BE,
     111X,23HADD. CAP.   INT.    FLOW,5X,6HSTRIKE/
     23X, 53HARC    FROM      TO     UNINT.   FLOW   REPAIR  REPAIR,
     33X, 58HFLOW   UNINT.  REPAIRED   TIMES   AT INT.    FLOW  AT INT.,
     42X,7HSUCCESS/
     52X, 53HNAME     NODE     NODE     COST     COST     COST     TIME,
     63X, 57HBOUND    COST    ON  DAY  STRUCK    COST        COST   COST,
     75X,5HPROB.)
  909 FORMAT(2H  ,3(A4,4X),   I5,4X,I5,2(2X,I6),2X,2(I5,3X),2X,I5,5X,I2,
     12X,2(4X,I5),3X,I5,5X,F4.2)
```

```
910 FORMAT(1H ,1X,4HDAY ,I3,9H MISSION ,I3,12H STRUCK ARC ,A4,
   116H, FROM LOCATION ,A4,13H TO LOCATION ,A4,30H.  IT WILL BE RESTOR
   2ED ON DAY ,I3,14H.  THE MISSION/2X,48HREDUCED TOTAL EXPECTED CAPAC
   3ITY ON THE ARC FROM ,I5, 8H UNITS (,I5,55H OF WHICH MAY FLOW AT TH
   4E UNINTERDICTED UNIT FLOW COST)/1X,4H TO ,I5,8H UNITS (,I5,96H OF
   5WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR
   6 COST FOR THE CAMPAIGN IS ,I8,1H,,/2X,52HAND THE TOTAL EXPECTED NE
   7TWORK FLOW COST PER DAY IS ,I8,18H.  THIS IS STRIKE ,I3,34H AGAINS
   8T THIS ARC.  THROUGHPUT IS ,I10,1H.////)
911 FORMAT(1H ,3X,3HARC,4X,4HFROM,4X,2HTO,3X,8HCAPACITY,2X,4HCOST,2X,
   14HFLOW,2X,15HREPAIRED ON DAY/)
912 FORMAT(1H ,///10X,27HFLOW TOTALLY STOPPED ON DAY  , I10,/// )
913 FORMAT(1H ,///10X,38HMINIMIUM THRU-PUT CANNOT BE MET ON DAY
   1I10,////)
915 FORMAT(1H , /// 32X,28HCOST OF THE PRESENT FLOW IS   ,I8,1H.
   2,5X,15H THROUGHPUT IS   ,I8,1H.////)
```

Appendix C

SAMPLE PROBLEM

For illustrative purposes a sample problem is solved on the combined road-rail network of Fig. 3. The sources are S and S', and the sinks are T and T'. It is assumed that the LOC user is attempting to maximize flow and that the individual arc parameters are given explicitly in the input deck.

Input Deck

```
                          SAMPLE ROAD-RAIL LOC NETWORK
   5     2     1     1     4     1   10    2     1
ARCS
  TTSS     TT     SS   1000    -1000      0        0        0      .0
  TTSS     TT     SS   1000    -1000      0        0        0      .0
  SSS      SS     S    1000        0      0        0        0      .0
  SSS      SS     S    1000        0      0        0        0      .0
  SSS'     SS     S'   1000        0      0        0        0      .0
  SSS'     SS     S'   1000        0      0        0        0      .0
  T TT     T      TT   1000        0      0        0        0      .0
  T TT     T      TT   1000        0      0        0        0      .0
  T'TT     T'     TT   1000        0      0        0        0      .0
  T'TT     T'     TT   1000        0      0        0        0      .0
  S A      S      A       8       10      6        2        0      .25
  S A      S      A       6       12      6        2        0      .25
  S A      A      S       8       10      6        2        0      .25
  S A      A      S       6       12      6        2        0      .25
  S C      S      C       4        5      2        2        0      .29
  S C      S      C       2        6      2        2        0      .29
  S C      C      S       4        5      2        2        0      .29
  S C      C      S       2        6      2        2        0      .29
  S D      S      D       5        7      5        2        0      .30
  S D      S      D       4       10      5        2        0      .30
  S D      D      S       5        7      5        2        0      .30
  S D      D      S       4       10      5        2        0      .30
  A B      A      B       7        5      7        1        0      .17
  A B      A      B       4        6      7        1        0      .17
  A B      B      A       7        5      7        1        0      .17
  A B      B      A       4        6      7        1        0      .17
  A T      A      T       9        7      5        1        0      .31
  A T      A      T       5        9      5        1        0      .31
  A T      T      A       9        7      5        1        0      .31
  A T      T      A       5        9      5        1        0      .31
  B C      B      C       8        7      6        2        0      .28
  B C      B      C       5        9      6        2        0      .28
  B C      C      B       8        7      6        2        0      .25
  B C      C      B       5        9      6        2        0      .25
  B T      B      T       9        7      6        2        0      .22
  B T      B      T       5        9      6        2        0      .22
  B T      T      B       9        7      6        2        0      .22
  B T      T      B       5        9      6        2        0      .22
  C D      C      D       7        6      5        1        0      .23
  C D      C      D       6        8      5        1        0      .23
  C D      D      C       7        6      5        1        0      .23
  C D      D      C       6        8      5        1        0      .23
  C T      C      T       8       10      3        1        0      .27
  C T      C      T       7       12      3        1        0      .27
  C T      T      C       8       10      3        1        0      .27
  C T      T      C       7       12      3        1        0      .27
  D T      D      T       4        6      4        3        0      .24
  D T      D      T       2        9      4        3        0      .24
  D T      T      D       4        6      4        3        0      .24
  D T      T      D       2        9      4        3        0      .24
  S S'     S      S'     15        2      5        1        0      .18
  S S'     S      S'     10        3      5        1        0      .18
  S S'     S'     S      15        2      5        1        0      .18
```

S S'	S'	S	10	3	5	1	0	.18
D D'	D	D'	18	2	5	1	0	.20
D D'	D	D'	12	3	5	1	0	.20
D D'	D'	D	18	2	5	1	0	.20
D D'	D'	D	12	3	5	1	0	.20
S'C'	S'	C'	12	3	8	2	0	.19
S'C'	S'	C'	4	6	8	2	0	.19
S'C'	C'	S'	12	3	8	2	0	.19
S'C'	C'	S'	4	6	8	2	0	.19
S'D'	S'	D'	11	4	7	3	0	.22
S'D'	S'	D'	3	6	7	3	0	.22
S'D'	D'	S'	11	4	7	3	0	.22
S'D'	D'	S'	3	6	7	3	0	.22
C'D'	C'	D'	9	4	6	2	0	.17
C'D'	C'	D'	3	7	6	2	0	.17
C'D'	D'	C'	9	4	6	2	0	.17
C'D'	D'	C'	3	7	6	2	0	.17
C'T'	C'	T'	10	3	7	2	0	.18
C'T'	C'	T'	2	5	7	2	0	.18
C'T'	T'	C'	10	3	7	2	0	.18
C'T'	T'	C'	2	5	7	2	0	.18
D'T'	D'	T'	13	3	6	2	0	.21
D'T'	D'	T'	2	5	6	2	0	.21
D'T'	T'	D'	13	3	6	2	0	.21
D'T'	T'	D'	2	5	6	2	0	.21

COMPUTE

Output Listing

The output listing is given on the following five pages.

-39-

UNCLASSIFIED

SAMPLE ROAD-RAIL LOC NETWORK

CAMPAIGN WILL LAST FOR 2 DAYS, WITH 5 MISSIONS PER DAY.

ARC NAME	FROM NODE	TO NODE	CAP. AT UNINT. COST	UNINT. FLOW COST	REPAIR COST	REPAIR TIME	LOWER FLOW BOUND	FLOW AT UNINT. COST	TO BE REPAIRED ON DAY	TIMES STRUCK	ADD. CAP. AT INT. COST	INT. FLOW COST	FLOW AT INT. COST	STRIKE SUCCESS PROB.
TTSS	TT	SS	1000	-1000	0	0	0	40	0	0	0	-1000	0	0.0
SSS	SS	S	1000	0	0	0	0	17	0	0	0	0	0	0.0
SSS'	SS	S'	1000	0	0	0	0	23	0	0	0	0	0	0.0
T TT	T	TT	1000	0	0	0	0	17	0	0	0	0	0	0.0
T'TT	T'	TT	1000	0	0	0	0	23	0	0	0	0	0	0.0
S A	S	A	8	10	2	2	0	8	0	0	0	12	0	0.25
S A	A	S	8	10	2	2	0	0	0	0	0	12	0	0.25
S C	S	C	4	5	1	2	0	4	0	0	0	6	0	0.29
S C	C	S	4	5	1	2	0	0	0	0	0	6	0	0.29
S D	S	D	5	7	1	2	0	5	0	0	0	10	0	0.30
S D	D	S	5	7	1	2	0	0	0	0	0	10	0	0.30
A B	A	B	7	5	1	1	0	0	0	0	0	6	0	0.17
A B	B	A	7	5	1	1	0	0	0	0	0	6	0	0.17
A T	A	T	9	7	2	1	0	8	0	0	0	9	0	0.31
A T	T	A	9	7	2	1	0	0	0	0	0	9	0	0.31
B C	B	C	8	7	2	2	0	0	0	0	0	9	0	0.28
B C	C	B	8	7	2	2	0	0	0	0	0	9	0	0.25
B T	B	T	9	7	1	2	0	0	0	0	0	9	0	0.22
B T	T	B	9	7	1	2	0	0	0	0	0	9	0	0.22
C D	C	D	7	6	1	1	0	0	0	0	0	8	0	0.23
C D	D	C	7	6	1	1	0	1	0	0	0	8	0	0.23
C T	C	T	8	10	1	1	0	5	0	0	0	12	0	0.27
C T	T	C	8	10	1	1	0	0	0	0	0	12	0	0.27
D T	D	T	4	6	1	3	0	4	0	0	0	9	0	0.24
D T	T	D	4	6	1	3	0	0	0	0	0	9	0	0.24
S S'	S	S'	15	2	1	1	0	0	0	0	0	3	0	0.18
S S'	S'	S	15	2	1	1	0	0	0	0	0	3	0	0.18
D D'	D	D'	18	2	1	1	0	0	0	0	0	3	0	0.20
D D'	D'	D	18	2	1	1	0	0	0	0	0	3	0	0.20
S'C'	S'	C'	12	3	2	2	0	12	0	0	0	6	0	0.19
S'C'	C'	S'	12	3	2	2	0	0	0	0	0	6	0	0.19
S'D'	S'	D'	11	4	2	3	0	11	0	0	0	6	0	0.22
S'D'	D'	S'	11	4	2	3	0	0	0	0	0	6	0	0.22
C'D'	C'	D'	9	4	1	2	0	2	0	0	0	7	0	0.17
C'D'	D'	C'	9	4	1	2	0	0	0	0	0	7	0	0.17
C'T'	C'	T'	10	3	1	2	0	10	0	0	0	5	0	0.18
C'T'	T'	C'	10	3	1	2	0	0	0	0	0	5	0	0.18
D'T'	D'	T'	13	3	1	2	0	13	0	0	0	5	0	0.21
D'T'	T'	D'	13	3	1	2	0	0	0	0	0	5	0	0.21

COST OF THE PRESENT FLOW IS 428. THROUGHPUT IS 40.

```
DAY   1 MISSION   1 STRUCK ARC S'D', FROM LOCATION   D' TO LOCATION   S'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM    11 UNITS (   11 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO    10 UNITS (    9 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS      2,
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS     412.  THIS IS STRIKE   1 AGAINST THIS ARC.  THROUGHPUT IS        39.

DAY   1 MISSION   1 STRUCK ARC S'D', FROM LOCATION   S' TO LOCATION   D'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM    11 UNITS (   11 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO    10 UNITS (    9 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS      2,
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS     412.  THIS IS STRIKE   1 AGAINST THIS ARC.  THROUGHPUT IS        39.

DAY   1 MISSION   2 STRUCK ARC S'D', FROM LOCATION   D' TO LOCATION   S'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM    10 UNITS (    9 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO     8 UNITS (    7 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS      3,
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS     396.  THIS IS STRIKE   2 AGAINST THIS ARC.  THROUGHPUT IS        37.

DAY   1 MISSION   2 STRUCK ARC S'D', FROM LOCATION   S' TO LOCATION   D'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM    10 UNITS (    9 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO     8 UNITS (    7 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS      3,
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS     396.  THIS IS STRIKE   2 AGAINST THIS ARC.  THROUGHPUT IS        37.

DAY   1 MISSION   3 STRUCK ARC S'D', FROM LOCATION   D' TO LOCATION   S'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM     8 UNITS (    7 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO     7 UNITS (    5 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS      4,
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS     390.  THIS IS STRIKE   3 AGAINST THIS ARC.  THROUGHPUT IS        36.

DAY   1 MISSION   3 STRUCK ARC S'D', FROM LOCATION   S' TO LOCATION   D'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM     8 UNITS (    7 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO     7 UNITS (    5 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS      4,
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS     390.  THIS IS STRIKE   3 AGAINST THIS ARC.  THROUGHPUT IS        36.

DAY   1 MISSION   4 STRUCK ARC S'D', FROM LOCATION   D' TO LOCATION   S'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM     7 UNITS (    5 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO     6 UNITS (    4 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS      5,
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS     382.  THIS IS STRIKE   4 AGAINST THIS ARC.  THROUGHPUT IS        35.

DAY   1 MISSION   4 STRUCK ARC S'D', FROM LOCATION   S' TO LOCATION   D'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM     7 UNITS (    5 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO     6 UNITS (    4 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS      5,
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS     382.  THIS IS STRIKE   4 AGAINST THIS ARC.  THROUGHPUT IS        35.
```

DAY 1 MISSION 5 STRUCK ARC S'D', FROM LOCATION D' TO LOCATION S'. [illegible] BE RESTORED ON DAY 4. THE MISSION REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM 6 UNITS (4 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST) TO 5 UNITS (3 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST). TOTAL REPAIR COST FOR THE CAMPAIGN IS 6. AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS 375. THIS IS STRIKE 5 AGAINST THIS ARC. THROUGHPUT IS 34.

DAY 1 MISSION 5 STRUCK ARC S'D', FROM LOCATION [illegible] LOCATION D'. IT WILL BE RESTORED ON DAY 4. THE MISSION REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM [illegible] (4 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST) TO 5 UNITS (3 OF WHICH MAY FLOW AT THE UNINT[illegible]CTED UNIT FLOW COST). TOTAL REPAIR COST FOR THE CAMPAIGN IS 6. AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS 375. THIS IS STRIKE 5 AGAINST THIS ARC. THROUGHPUT IS 34.

PROFILE POINT	FLOW	COST	PROFILE POINT	FLOW	COST
1	3	18			
2	7	42			
3	10	60			
4	14	90			
5	17	119			
6	20	155			
7	24	209			
8	27	256			
9	31	324			
10	34	375			

ARC NAME	FROM NODE	TO NODE	CAP. AT UNINT. COST	UNINT. FLOW COST	REPAIR COST	REPAIR TIME	LOWER FLOW BOUND	FLOW AT UNINT. COST	TO BE REPAIRED ON DAY	TIMES STRUCK	ADD. CAP. AT INT. COST	INT. FLOW COST	FLOW AT INT. COST	STRIKE SUCCESS PROB.
TTSS	TT	SS	1000	-1000	0	0	0	34	0	0	0	-1000	0	0.0
SSS	SS	S	1000	0	0	0	0	17	0	0	0	0	0	0.0
SSS'	SS	S'	1000	0	0	0	0	17	0	0	0	0	0	0.0
T TT	T	TT	1000	0	0	0	0	12	0	0	0	0	0	0.0
T'TT	T'	TT	1000	0	0	0	0	22	0	0	0	0	0	0.0
S A	S	A	8	10	2	2	0	8	0	0	0	12	0	0.25
S C	S	C	4	5	1	2	0	4	0	0	0	6	0	0.29
S D	S	D	5	7	1	2	0	5	0	0	0	10	0	0.30
A T	A	T	9	7	2	1	0	8	0	0	0	9	0	0.31
C T	C	T	8	10	1	1	0	4	0	0	0	12	0	0.27
D D'	D	D'	18	2	1	1	0	5	0	0	0	3	0	0.20
S'C'	S'	C'	12	3	2	2	0	12	0	0	0	6	0	0.19
S'D'	S'	D'	3	4	0	3	0	3	4	5	2	6	2	0.22
C'D'	C'	D'	9	4	1	2	0	2	0	0	0	7	0	0.17
C'T'	C'	T'	10	3	1	2	0	10	0	0	0	5	0	0.18
D'T'	D'	T'	13	3	1	2	0	12	0	0	0	5	0	0.21

COST OF THE PRESENT FLOW IS 375. THROUGHPUT IS 34.

```
DAY   2 MISSION   1 STRUCK ARC S'D', FROM LOCATION   D' TO LOCATION   S'.  IT WILL BE RESTORED ON DAY   5.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM     5 UNITS (    3 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO     4 UNITS (    2 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS        6.
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS       368.  THIS IS STRIKE   6 AGAINST THIS ARC.  THROUGHPUT IS        33.

DAY   2 MISSION   1 STRUCK ARC S'D', FROM LOCATION   S' TO LOCATION   D'.  IT WILL BE RESTORED ON DAY   5.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM     5 UNITS (    3 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO     4 UNITS (    2 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS        6.
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS       368.  THIS IS STRIKE   6 AGAINST THIS ARC.  THROUGHPUT IS        33.

DAY   2 MISSION   2 STRUCK ARC S'C', FROM LOCATION   C' TO LOCATION   S'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM    12 UNITS (   12 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO    11 UNITS (   10 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS        8.
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS       361.  THIS IS STRIKE   1 AGAINST THIS ARC.  THROUGHPUT IS        32.

DAY   2 MISSION   2 STRUCK ARC S'C', FROM LOCATION   S' TO LOCATION   C'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM    12 UNITS (   12 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO    11 UNITS (   10 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS        8.
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS       361.  THIS IS STRIKE   1 AGAINST THIS ARC.  THROUGHPUT IS        32.

DAY   2 MISSION   3 STRUCK ARC S'C', FROM LOCATION   C' TO LOCA[illegible]N   S'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM    11 UNITS (   [illegible] OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO     9 UNITS (    8 OF WHICH MAY FLOW AT THE UNINTERDICTED UN[illegible] FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS        9.
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS       345.  THIS IS STRIKE   2 AGAINST THIS ARC.  THROUGHPUT IS        30.

DAY   2 MISSION   3 STRUCK ARC S'C', FROM LOCATION   S' TO LOCATION   C'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM    11 UNITS (   10 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO     9 UNITS (    8 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS        9.
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS       345.  THIS IS STRIKE   2 AGAINST THIS ARC.  THROUGHPUT IS        30.

DAY   2 MISSION   4 STRUCK ARC S'C', FROM LOCATION   C' TO LOCATION   S'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM     9 UNITS (    8 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO     8 UNITS (    6 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS       10.
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS       342.  THIS IS STRIKE   3 AGAINST THIS ARC.  THROUGHPUT IS        29.

DAY   2 MISSION   4 STRUCK ARC S'C', FROM LOCATION   S' TO LOCATION   C'.  IT WILL BE RESTORED ON DAY   4.  THE MISSION
REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM     9 UNITS (    8 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST)
TO     8 UNITS (    6 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST).  TOTAL REPAIR COST FOR THE CAMPAIGN IS       10.
AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS       342.  THIS IS STRIKE   3 AGAINST THIS ARC.  THROUGHPUT IS        29.
```

DAY 2 MISSION 5 STRUCK ARC S'C', FROM LOCATION C' TO LOCATION S'. IT WILL BE RESTORED ON DAY 4. THE MISSION REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM 8 UNITS (6 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST) TO 7 UNITS (5 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST). TOTAL REPAIR COST FOR THE CAMPAIGN IS 11. AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS 336. THIS IS STRIKE 4 AGAINST THIS ARC. THROUGHPUT IS 28.

DAY 2 MISSION 5 STRUCK ARC S'C', FROM LOCATION S' TO LOCATION C'. IT WILL BE RESTORED ON DAY 4. THE MISSION REDUCED TOTAL EXPECTED CAPACITY ON THE ARC FROM 8 UNITS (6 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST. TO 7 UNITS (5 OF WHICH MAY FLOW AT THE UNINTERDICTED UNIT FLOW COST). TOTAL REPAIR COST FOR THE CAMPAIGN IS 11. AND THE TOTAL EXPECTED NETWORK FLOW COST PER DAY IS 336. THIS IS STRIKE 4 AGAINST THIS ARC. THROUGHPUT IS 28.

ARC NAME	FROM NODE	TO NODE	CAP. AT UNINT. COST	UNINT. FLOW COST	REPAIR COST	REPAIR TIME	LOWER FLOW BOUND	FLOW AT UNINT. COST	TO BE REPAIRED ON DAY	TIMES STRUCK	ADD. CAP. AT INT. COST	INT. FLOW COST	FLOW AT INT. COST	STRIKE SUCCESS PROB.
TTSS	TT	SS	1000	-1000	0	0	0	28	0	0	0	-1000	0	0.0
SSS	SS	S	1000	0	0	0	0	17	0	0	0	0	0	0.0
SSS'	SS	S'	1000	0	0	0	0	11	0	0	0	0	0	0.0
T TT	T	TT	1000	0	0	0	0	12	0	0	0	0	0	0.0
T'TT	T'	TT	1000	0	0	0	0	16	0	0	0	0	0	0.0
S A	S	A	8	10	2	2	0	8	0	0	0	12	0	0.25
S C	S	C	4	5	1	2	0	4	0	0	0	6	0	0.29
S D	S	D	5	7	1	2	0	5	0	0	0	10	0	0.30
A T	A	T	5	7	2	1	0	8	0	0	0	10	0	0.31
C T	C	T	8	10	1	1	0	4	0	0	0	9	0	0.31
D D'	D	D'	18	2	1	1	0	5	0	0	0	12	0	0.27
S'C'	S'	C'	5	3	1	2	0	5	0	0	0	3	0	0.20
S'D'	S'	D'	2	4	0	3	0	2	4	4	2	6	2	0.19
C'T'	C'	T'	10	3	1	2	0	7	5	6	2	6	2	0.22
D'T'	D'	T'	13	3	1	2	0	9	0	0	0	5	0	0.18

COST OF THE PRESENT FLOW IS 375. THROUGHPUT IS 28.

PROFILE POINT	FLOW	COST	PROFILE POINT	FLOW	COST
1	3	18			
2	6	37			
3	8	53			
4	11	80			
5	14	116			
6	17	155			
7	20	203			
8	22	234			
9	25	265			
10	28	336			

REFERENCES

1. Durbin, E. P., An Interdiction Model of Highway Transportation, The RAND Corporation, RM-4945-PR, May 1966.
2. Ford, L. R., Jr., and D. R. Fulkerson, Flows in Networks, Princeton University Press, Princeton, New Jersey, 1963.
3. Fulkerson, D. R., An Out-of-Kilter Method for Minimal Cost Flow Problems, The RAND Corporation, P-1825, April 1960. Also published in the J. Soc. Indus. Appl. Math, Vol. 9, March 1961, pp. 18-37.
4. Wollmer, R. D., An Interdiction Model for Sparsely Traveled Networks, The RAND Corporation, RM-5539-PR, April 1968.
5. -----, "Removing Arcs From a Network," J. Op. Res. Soc. Am., 12, 934-40, 1964.
6. -----, Some Methods for Determining the Most Vital Link in a Railway Network, The RAND Corporation, RM-3321-ISA, April 1963.
7. -----, Algorithms for Targeting Strikes in an LOC Network, The RAND Corporation, RM-5864-PR, February 1969.

DOCUMENT CONTROL DATA

1. ORIGINATING ACTIVITY THE RAND CORPORATION	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP
3. REPORT TITLE A MODEL FOR TARGETING STRIKES IN AN LOC NETWORK	
4. AUTHOR(S) (Last name, first name, initial) Wollmer, R. D. and M. J. Ondrasek	

5. REPORT DATE September 1969	6a. TOTAL NO. OF PAGES 54	6b. NO. OF REFS. 7
7. CONTRACT OR GRANT NO. F44620-67-C-0045	8. ORIGINATOR'S REPORT NO. RM-5940-PR	

9a. AVAILABILITY/LIMITATION NOTICES DDC-1	9b. SPONSORING AGENCY United States Air Force Project RAND

10. ABSTRACT

A computer model for developing and evaluating a targeting strategy against an opposing force's lines of communication. This aim is to obtain the greatest reduction in enemy throughput and the greatest time and cost of repair. The network arcs (road, rail, or waterway segments or transshipment points) are characterized by beginning and ending nodes, upper and lower bounds, interdicted and uninterdicted unit flow costs, repair times and costs, and the probabilities that attempted strikes are successful. The model is programmed in daily cycles, with the user specifying number of days and strikes. Strikes are targeted one by one. At the end of each strike, total LOC throughput and costs are printed out; if desired, a detailed status report and/or a profile of total flow versus user cost are also output. The FORTRAN program is thoroughly self-documented.

11. KEY WORDS

Interdiction
Computer programs
Networks
Models
Logistics
Transportation
Targets
Counterinsurgency and insurgency
Lines of Communication (LOC) Model
Mathematics